PRESIDENCY UNIVERSITY

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

PROGRAMME REGULATIONS & CURRICULUM

2023-27

PRESIDENCY SCHOOL OF COMPUTER SCIENCE & ENGINEERING

BACHELOR OF TECHNOLOGY (B.TECH.)

COMPUTER SCIENCE AND ENGINEERING

e

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Program Regulations and Curriculum

2023-2027

Bachelor of Technology (B.Tech.) in Computer Science and Engineering

based on Choice Based Credit System (CBCS) and Outcome Based Education (OBE)

(As amended up to the 24th Meeting of the Academic Council held on 3rd August 2024. This document supersedes all previous guidelines)

Regulations No.: PU/AC-24.5/SOCSE04/CSE/2023-2027

AUGUST - 2024

Resolution No. 5 of the 24th Meeting of the Academic Council held on 3rd August 2024, and ratified by the Board of Management in its 24th Meeting held on 5th August 2024

Table of Contents

Clause No.	Contents	Page Number
PART A – PROGRAM REGULATIONS		
1.	Vision & Mission of the University and the School / Department	3
2.	Preamble to the Program Regulations and Curriculum	3
3.	Short Title and Applicability	3
4.	Definitions	4
5.	Program Description	5
6.	Minimum and Maximum Duration	6
7.	Programme Educational Objectives (PEO)	7
8.	Programme Outcomes (PO) and Programme Specific Outcomes (PSO)	7
9.	Admission Criteria (as per the concerned Statutory Body)	8
10.	Lateral Entry / Transfer Students requirements	9
11.	Change of Branch / Discipline / Specialization	11
12.	Specific Regulations regarding Assessment and Evaluation	12
13.	Additional clarifications - Rules and Guidelines for Transfer of Credits from MOOC, etc.	14
PART B: PROGRAM STRUCTURE		
14.	Structure / Component with Credit Requirements Course Baskets & Minimum Basket wise Credit Requirements	16
15.	Minimum Total Credit Requirements of Award of Degree	17
16.	Other Specific Requirements for Award of Degree, if any, as prescribed by the Statutory Bodies	17
PART C: CURRICULUM STRUCTURE		
17.	Curriculum Structure – Basket Wise Course List	18
18.	Practical / Skill based Courses – Internships / Thesis / Dissertation / Capstone Project Work / Portfolio / Mini project	20
19.	List of Elective Courses under various Specializations / Stream Basket	22
20.	List of Open Electives to be offered by the School / Department (Separately for ODD and EVEN Semesters).	27
21.	List of MOOC (NPTEL) Courses	31
22.	Recommended Semester Wise Course Structure / Flow including the Program / Discipline Elective Paths / Options	32
23.	Course Catalogue of all Courses Listed including the Courses Offered by other School / Department and Discipline / Program Electives	36

PART A – PROGRAM REGULATIONS

1. Vision & Mission of the University and the School / Department

1.1 Vision of the University

To be a Value-driven Global University, excelling beyond peers and creating professionals of integrity and character, having concern and care for society.

1.2 Mission of the University

- Commit to be an innovative and inclusive institution by seeking excellence in teaching, research and knowledge-transfer.
- Pursue Research and Development and its dissemination to the community, at large.
- Create, sustain and apply learning in an interdisciplinary environment with consideration for ethical, ecological and economic aspects of nation building.
- Provide knowledge-based technological support and services to the industry in its growth and development.
- To impart globally-applicable skill-sets to students through flexible course offerings and support industry's requirement and inculcate a spirit of new-venture creation.

1.3 Vision of Presidency School of Computer Science and Engineering

To be a value based, practice-driven School of Computer Science and Engineering, committed to developing globally-competent Engineers, dedicated to transforming Society.

1.4 Mission of Presidency School of Computer Science and Engineering

- Cultivate a practice-driven environment with a contemporary Learning-pedagogy, integrating theory and practice.
- Attract and nurture world-class faculty to excel in Teaching and Research, in the field of Core Computer Science and Engineering.
- Establish state-of-the-art facilities for effective Teaching and Learning-experiences.
- Promote Interdisciplinary Studies to nurture talent and impart relevant skill-sets for global impact.
- Instil Entrepreneurial and Leadership Skills to address Social, Environmental, and Community-needs.

2. Preamble to the Program Regulations and Curriculum

This is the subset of Academic Regulations and it is to be followed as a requirement for the award of B.Tech degree.

The Curriculum is designed to take into the factors listed in the Choice Based Credit System (CBCS) with focus on Industrial Based Project Learning, Industrial Training, and Internship to enable the students to become eligible and fully equipped for employment in industries, choose higher studies or entrepreneurship.

In exercise of the powers conferred by and in discharge of duties assigned under the relevant provision(s) of the Act, Statutes and Academic Regulations, 2025 of the University, the Academic Council hereby makes the following Regulations.

3. Short Title and Applicability

a. These Regulations shall be called the Bachelor of Technology Degree Program Regulations and Curriculum 2023-2027.
b. These Regulations are subject to, and pursuant to the Academic Regulations.
c. These Regulations shall be applicable to the ongoing Bachelor of Technology Degree Programs of the 2023-2027 batch, and to all other Bachelor of Technology Degree Programs which may be introduced in future.
d. These Regulations shall supersede all the earlier Bachelor of Technology Degree Program Regulations and Curriculum, along with all the amendments thereto.
e. These Regulations shall come into force from the Academic Year 2024-2025.

4. Definitions

In these Regulations, unless the context otherwise requires:

a. "Academic Calendar" means the schedule of academic and miscellaneous events as approved by the Vice Chancellor;
b. "Academic Council" means the Academic Council of the University;

c. *"Academic Regulations" means the Academic Regulations, of the University;*

d. *"Academic Term" means a Semester or Summer Term;*

e. *"Act" means the Presidency University Act, 2013;*

f. *"AICTE" means All India Council for Technical Education;*

g. *"Basket" means a group of courses bundled together based on the nature/type of the course;*

h. *"BOE" means the Board of Examinations of the University;*

i. *"BOG" means the Board of Governors of the University;*

j. *"BOM" means the Board of Management of the University;*

k. *"BOS" means the Board of Studies of a particular Department/Program of Study of the University;*

l. *"CGPA" means Cumulative Grade Point Average as defined in the Academic Regulations;*

m. *"Clause" means the duly numbered Clause, with Sub-Clauses included, if any, of these Regulations;*

n. *"COE" means the Controller of Examinations of the University;*

o. *"Course In Charge" means the teacher/faculty member responsible for developing and organising the delivery of the Course;*

p. *"Course Instructor" means the teacher/faculty member responsible for teaching and evaluation of a Course;*

q. *"Course" means a specific subject usually identified by its Course-code and Course-title, with specified credits and syllabus/course-description, a set of references, taught by some teacher(s)/course-instructor(s) to a specific class (group of students) during a specific Academic Term;*

r. *"Curriculum Structure" means the Curriculum governing a specific Degree Program offered by the University, and, includes the set of Baskets of Courses along with minimum credit requirements to be earned under each basket for a degree/degree with specialization/minor/honours in addition to the relevant details of the Courses and Course catalogues (which describes the Course content and other important information about the Course). Any specific requirements for a particular program may be brought into the Curriculum structure of the specific program and relevant approvals should be taken from the BOS and Academic Council at that time.*

s. *"DAC" means the Departmental Academic Committee of a concerned Department/Program of Study of the University;*

t. *"Dean" means the Dean / Director of the concerned School;*

u. *"Degree Program" includes all Degree Programs;*

v. *"Department" means the Department offering the degree Program(s) / Course(s) / School offering the concerned Degree Programs / other Administrative Offices;*

w. *"Discipline" means specialization or branch of B.Tech. Degree Program;*

x. *"HOD" means the Head of the concerned Department;*

y. *"L-T-P-C" means Lecture-Tutorial-Practical-Credit – refers to the teaching – learning periods and the credit associated;*

z. *"MOOC" means Massive Open Online Courses;*

aa. *"MOU" means the Memorandum of Understanding;*

bb. *"NPTEL" means National Program on Technology Enhanced Learning;*

cc. *"Parent Department" means the department that offers the Degree Program that a student undergoes;*

dd. *"Program Head" means the administrative head of a particular Degree Program/s;*

ee. *"Program Regulations" means the Bachelor of Technology Degree Program Regulations and Curriculum, 2024-2028;*

ff. *"Program" means the Bachelor of Technology (B.Tech.) Degree Program;*

gg. *"PSCS" means the Presidency School of Computer Science;*

hh. *"Registrar" means the Registrar of the University;*

ii. *"School" means a constituent institution of the University established for monitoring, supervising and guiding, teaching, training and research activities in broadly related fields of studies;*

jj. *"Section" means the duly numbered Section, with Clauses included in that Section, of these Regulations;*

kk. *"SGPA" means the Semester Grade Point Average as defined in the Academic Regulations, 2021;*

ll. *"Statutes" means the Statutes of Presidency University;*

mm. *"Sub-Clause" means the duly numbered Sub-Clause of these Program Regulations;*

nn. *"Summer Term" means an additional Academic Term conducted during the summer break (typically in June-July) for a*

duration of about eight (08) calendar weeks, with a minimum of thirty (30) University teaching days;

oo. "SWAYAM" means Study Webs of Active Learning for Young Aspiring Minds.

pp. "UGC" means University Grant Commission;

qq. "University" means Presidency University, Bengaluru; and

rr. "Vice Chancellor" means the Vice Chancellor of the University.

5. Program Description

The Bachelor of Technology Degree Program Regulations and Curriculum 2023-2027 are subject to, and, pursuant to the Academic Regulations. These Program Regulations shall be applicable to the following ongoing Bachelor of Technology (B.Tech.) Degree Programs of 2023-2027 offered by the Presidency School of Computer Science and Engineering (PSCS):

1. Bachelor of Technology in Computer Science and Engineering, abbreviated as B.Tech. Computer Science and Engineering;
2. Bachelor of Technology in Computer Science and Technology (Big Data), abbreviated as B.Tech. Computer Science and Technology (Big Data);
3. Bachelor of Technology in Computer Science and Engineering (Block Chain), abbreviated as B.Tech. Computer Science and Engineering (Block Chain);
4. Bachelor of Technology in Computer Science and Technology (Dev Ops), abbreviated as B.Tech. Computer Science and Technology (Dev Ops);
5. Bachelor of Technology in Computer Science and Engineering (Cyber Security), abbreviated as B.Tech. Computer Science and Engineering (Cyber Security);
6. Bachelor of Technology in Computer Science and Engineering (Internet of Things), abbreviated as B.Tech. Computer Science and Engineering (Internet of Things);
7. Bachelor of Technology in Computer Science and Engineering (Data Science), abbreviated as B.Tech. Computer Science and Engineering (Data Science);
8. Bachelor of Technology in Computer Science and Technology (Artificial Intelligence and Machine Learning), abbreviated as B.Tech. Computer Science and Technology (Artificial Intelligence and Machine Learning);
9. Bachelor of Technology in Information Science and Technology, abbreviated as B.Tech. Information Science and Technology;
10. Bachelor of Technology in Computer Science and Information Technology, abbreviated as B.Tech. Computer Science and Information Technology;
11. Bachelor of Technology in Computer Science and Engineering (Networks), abbreviated as B.Tech. Computer Science and Engineering (Networks);
12. Bachelor of Technology in Computer Engineering (Artificial Intelligence and Machine Learning), abbreviated as B.Tech. Computer Engineering (Artificial Intelligence and Machine Learning);
13. Bachelor of Technology in Information Science and Engineering (Artificial Intelligence and Robotics), abbreviated as B.Tech. Information Science and Engineering (Artificial Intelligence and Robotics); and
14. Bachelor of Technology in Computer Science and Engineering (Artificial Intelligence and Machine Learning) abbreviated as B.Tech. Computer Science and Engineering (Artificial Intelligence and Machine Learning);

5.1. Bachelor of Technology in Computer Science and Engineering, abbreviated as B.Tech. (Computer Science and Engineering)

5.2 These Program Regulations shall be applicable to other similar programs, which may be introduced in future.

5.3 These Regulations may evolve and get amended or modified or changed through appropriate approvals from the Academic Council, from time to time, and shall be binding on all concerned.

6. Minimum and Maximum Duration

6.1 Bachelor of Technology Degree Program is a Four-Year, Full-Time Semester based program. The minimum duration of the B.Tech. Program is four (04) years and each year comprises of two academic Semesters (Odd and Even Semesters) and hence the duration of the B.Tech. program is eight (08) Semesters.

6.2 A student who for whatever reason is not able to complete the Program within the normal period or the minimum duration (number of years) prescribed for the Program, may be allowed a period of two years beyond the normal period to complete the mandatory minimum credits requirement as prescribed by the concerned Program Regulations and Curriculum. In general, the permissible maximum duration (number of years) for completion of Program is 'N' + 2 years, where 'N' stands for the normal or minimum duration (number of years) for completion of the concerned Program as prescribed by the concerned Program Regulations and Curriculum.

6.3 The time taken by the student to improve Grades/CGPA, and in case of temporary withdrawal/re-joining (Refer to Clause 16.1 in Academic Regulations), shall be counted in the permissible maximum duration for completion of a Program.

6.4 In exceptional circumstances, such as temporary withdrawal for medical exigencies where there is a prolonged hospitalization and/or treatment, as certified through hospital/medical records, women students requiring extended maternity break (certified by registered medical practitioner), and, outstanding sportspersons representing the University/State/India requiring extended time to participate in National/International sports events, a further extension of one (01) year may be granted on the approval of the Academic Council.

6.5 The enrolment of the student who fails to complete the mandatory requirements for the award of the concerned Degree (refer Section 19.0 of Academic Regulations) in the prescribed maximum duration (Sub-Clauses 18.1 and 18.2 of Academic Regulations), shall stand terminated and no Degree shall be awarded.

7 Programme Educational Objectives (PEO)

After four years of successful completion of the program, the graduates shall be able to:

PEO1. Demonstrate as a Computer Engineering Professional with innovative skills and moral and ethical values.

PEO2. Become a Teaching and Research Professional in the area of Computer science and engineering through lifelong learning.

PEO3. Emerge as a Consultancy team member in the Computer Science and Engineering Industry.

PEO4. Evolve as an entrepreneur in the computer science and other related areas of specialization.

8 Programme Outcomes (PO) and Programme Specific Outcomes (PSO)

8.1 Programme Outcomes (PO)

On successful completion of the Program, the students shall be able to:

PO1. Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2. Problem Analysis: Identify, formulate, review research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3. **Design/Development of Solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4. Conduct Investigations of Complex Problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5. Modern Tool Usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6. The Engineer and Society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7. Environment and Sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9. Individual and Team Work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11. Project Management and Finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12. Life-Long Learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

8.2 Program Specific Outcomes (PSOs):

On successful completion of the Program, the students shall be able to:

PSO1: Problem Analysis: Identify, formulate, research literature, and analyze complex engineering problems related to Software Engineering principles and practices, Programming and Computing technologies reaching substantiated conclusions using first principle

PSO2: Design/development of Solutions: Design solutions for complex engineering problems related to Software Engineering principles and practices, Programming and Computing technologies and design system components or processes that meet the specified needs

PSO3: Modern Tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities related to Software Engineering principles and practices, Programme.

9 Admission Criteria (as per the concerned Statutory Body)

The University admissions shall be open to all persons irrespective of caste, class, creed, gender or nation. All admissions shall be made on the basis of merit in the qualifying examinations; provided that forty percent of the admissions in all Programs of the University shall be reserved for the students of Karnataka State and admissions shall be made through a Common Entrance Examination conducted by the State Government or its agency and seats shall be allotted as per the merit and reservation policy of the State Government from time to time**.** The admission criteria to the B.Tech. Program is listed in the following Sub-Clauses:

9.1 An applicant who has successfully completed Pre-University course or Senior Secondary School course (+2) or equivalent such as (11+1), 'A' level in Senior School Leaving Certificate Course from a recognized university of India or outside or from Senior Secondary Board or equivalent, constituted or recognized by the Union or by the State Government of that Country for the purpose of issue of qualifying certificate on successful completion of the course, may apply for and be admitted into the Program.

9.2 Provided further, the applicant must have taken Physics and Mathematics as compulsory subjects in the Pre-University / Higher Secondary / (10+2) / (11+1) examination, along with either Chemistry / Biology / Electronics / Computer Science / Biotechnology subject, and, the applicant must have obtained a minimum of 45% of the total marks (40% in case of candidates belonging to the Reserved Category as classified by the Government of Karnataka) in these subjects taken together.

9.3 The applicant must have appeared for Joint Entrance Examinations (JEE) Main / JEE (Advanced) / Karnataka CET / COMED-K, or any other State-level Engineering Entrance Examinations.

9.4 Reservation for the SC / ST and other backward classes shall be made in accordance with the directives issued by the

Government of Karnataka from time to time.

9.5 Admissions are offered to Foreign Nationals and Indians living abroad in accordance with the rules applicable for such admission, issued from time to time, by the Government of India.

9.6 Candidates must fulfil the medical standards required for admission as prescribed by the University.

9.7 If, at any time after admission, it is found that a candidate had not in fact fulfilled all the requirements stipulated in the offer of admission, in any form whatsoever, including possible misinformation and any other falsification, the Registrar shall report the matter to the Board of Management (BOM), recommending revoking the admission of the candidate.

9.8 The decision of the BOM regarding the admissions is final and binding.

10 Lateral Entry / Transfer Students requirements

10.1 Lateral Entry

The University admits students directly to the second year (3rd Semester) of the B.Tech. Degree program as per the provisions and/or regulations of the Government of Karnataka pertaining to the "Lateral Entry" scheme announced by the Government from time to time. Further, the general conditions and rules governing the provision of Lateral Entry to the B.Tech. Program of the University are listed in the following Sub-Clauses:

10.1.1 Admission to 2nd year (3rd Semester) of the B.Tech. Degree program shall be open to the candidates who are holders of a 3-year Diploma in Engineering (or equivalent qualification as recognized by the University), who have secured not less than forty-five percentage (45%) marks in the final year examination (5th and 6th Semesters of the Diploma Program) in the appropriate branch of Engineering. Provided that, in case of SC / ST and OBC candidates from Karnataka the minimum marks for eligibility shall be forty percent (40%).

10.1.2 Provided further that, candidates seeking Lateral Entry may be required to complete specified bridge Courses as prescribed by the University. Such bridge Courses, if any, shall not be included in the CGPA computations.

10.1.3 All the existing Regulations and Policies of the University shall be binding on all the students admitted to the Program through the provision of Lateral Entry.

10.1.4 The Course requirements prescribed for the 1st Year of the B.Tech. Program shall be waived for the student(s) admitted through Lateral Entry and the duration of the B.Tech. Program for such students is three (03) years, commencing from the 3rd Semester (commencement of the 2nd Year) of the B.Tech. Program and culminating with the 8th Semester (end of the 4th Year) of the B.Tech. Program.

10.1.5 Provided that, if a Lateral Entry student misses any mandatory program specific courses that are typically offered in the 1st year (1st or 2nd semesters), then those courses must be cleared by the students as soon as possible, preferably during the Summer Term.

10.1.6 The existing Program Regulations of the concerned Program to which the student is admitted through the provision of Lateral Entry shall be binding on the student with effect from the 3rd Semester of the Program. i.e., the Program Structure and Curriculum from the 3rd to 8th Semesters of the Program concerned shall be binding on the student admitted through Lateral Entry. Further, any revisions / amendments made to the Program Regulations thereafter, shall be binding on all the students of the concerned Program.

10.1.7 All the Courses (and the corresponding number of Credits) prescribed for the 1st Year of the concerned B.Tech. Program shall be waived for the student(s) admitted to the concerned B.Tech Program through Lateral Entry. Further, the *Minimum Credit Requirements* for the award of the B.Tech. Degree in the concerned Program shall be prescribed / calculated as follows:

The ***Minimum Credit Requirements*** for the award of the Bachelor of Technology (B.Tech.) Degree prescribed by the concerned Bachelor of Technology Degree Program Regulations and Curriculum, 2024-2028, minus the number of Credits prescribed / accepted by the Equivalence Committee for the 1st Year (1st and 2nd Semesters) of the B.Tech. Program.

For instance, if the *Minimum Credit Requirements* for the award of the Bachelor of Technology (B.Tech.) Degree as prescribed by the Regulations for B.Tech. (Computer Science and Engineering) is "N" Credits, and, if the total credits prescribed in the 1st Year (total credits of the 1st and 2nd Semesters) of the Program concerned is "M" Credits, then the *Minimum Credit Requirements* for the award of the B.Tech. in Computer Science and Engineering for a student who joins the Program through the provision of the Lateral Entry, shall be "N – M" Credits.

10.1.8 Further, no other waiver except the Courses prescribed for the 1st year of the B.Tech. Program of the University shall be permissible for students joining the B.Tech. Program through the provision of Lateral Entry.

10.2 Transfer of student(s) from another recognized University to the 2nd year (3rd Semester) of the B.Tech. Program of the University

A student who has completed the 1st Year (i.e., passed in all the Courses / Subjects prescribed for the 1st Year) of the B.Tech./B.E./B.S., Four-Year Degree Program from another recognized University, may be permitted to transfer to the 2nd Year (3rd Semester) of the B.Tech. Program of the Presidency University as per the rules and guidelines prescribed in the following Sub-Clauses:

10.2.1 The concerned student fulfils the criteria specified in Sub-Clauses 10.1.1, 10.1.2 and 10.1.3.

10.2.2 The student shall submit the Application for Transfer along with a non-refundable Application Fee (as prescribed by the University from time to time) to the Presidency University no later than July 10 of the concerned year for admission to the 2nd Year (3rd Semester) B.Tech. Program commencing on August 1 on the year concerned.

10.2.3 The student shall submit copies of the respective Marks Cards / Grade Sheets / Certificates along with the Application for Transfer.

10.2.4 The transfer may be provided on the condition that the Courses and Credits completed by the concerned student in the 1st Year of the B.Tech./ B.E. / B.S. Four Degree Program from the concerned University, are declared equivalent and acceptable by the Equivalence Committee constituted by the Vice Chancellor for this purpose. Further, the Equivalence Committee may also prescribe the Courses and Credits the concerned students shall have to mandatorily complete, if admitted to the 2nd Year of the B.Tech. Program of the University.

10.2.5 The Branch / Discipline allotted to the student concerned shall be the decision of the University and binding on the student.

11 Change of Branch / Discipline / Specialization

A student admitted to a particular Branch of the B.Tech. Program will normally continue studying in that Branch till the completion of the program. However, the University reserves the right to provide the option for a change of Branch, or not to provide the option for a change of Branch, at the end of 1st Year of the B.Tech. Program to eligible students in accordance with the following rules and guidelines: framed by the University from time to time.

11.1 Normally, only those students, who have passed all the Courses prescribed for the 1st Year of the B.Tech. Program and obtained a CGPA of not less than 6.50 at the end of the 2nd Semester, shall be eligible for consideration for a change of Branch.

11.2 Change of Branch, if provided, shall be made effective from the commencement of the 3rd Semester of the B.Tech. Program. There shall be no provision for change of Branch thereafter under any circumstances whatsoever.

11.3 The student provided with the change of Branch shall fully adhere to and comply with the Program Regulations of the concerned Branch of the B.Tech. Program, the Fee Policy pertaining to that Branch of the B.Tech. Program, and, all other rules pertaining to the changed Branch existing at the time.

11.4 Change of Branch once made shall be final and binding on the student. No student shall be permitted, under any circumstances, to refuse the change of Branch offered.

11.5 The eligible student may be allowed a change in Branch, strictly in order of *inter se* merit, subject to the conditions given below:

11.5.1 The actual number of students in the 3rd Semester in any particular Branch to which the transfer is to be made, should not exceed the intake fixed by the University for the concerned Branch;

11.5.2 The actual number of students in any Branch from which transfer is being sought does not fall below 75% of the total intake fixed by the University for the concerned Branch.

The process of change of Branch shall be completed within the first five days of Registration for the 3rd Semester of the B.Tech. Program.

12 Specific Regulations regarding Assessment and Evaluation (including the Assessment Details of NTCC Courses, Weightages of Continuous Assessment and End Term Examination for various Course Categories)

12.1 The academic performance evaluation of a student in a Course shall be according to the University Letter Grading System based on the class performance distribution in the Course.

12.2 Academic performance evaluation of every registered student in every Course registered by the student is carried out through various components of Assessments spread across the Semester. The nature of components of Continuous Assessments and the weightage given to each component of Continuous Assessments (refer Clause 8.8 of Academic regulations) shall be clearly defined in the Course Plan for every Course, and approved by the DAC.

12.3 Format of the End-Term examination shall be specified in the Course Plan.

12.4 Grading is the process of rewarding the students for their overall performance in each Course. The University follows the system of Relative Grading with statistical approach to classify the students based on the relative performance of the students registered in the concerned Course except in the following cases:

- Non-Teaching Credit Courses (NTCC)
- Courses with a class strength less than 30

Absolute grading method may be adopted, where necessary with prior approval of concerned DAC.

Grading shall be done at the end of the Academic Term by considering the aggregate performance of the student in all components of Assessments prescribed for the Course. Letter Grades (Clause 8.10 of Academic regulations) shall be awarded to a student based on her/his overall performance relative to the class performance distribution in the concerned Course. These Letter Grades not only indicate a qualitative assessment of the student's performance but also carry a quantitative (numeric) equivalent called the Grade Point.

12.5 Assessment Components and Weightage

Table 1: Assessment Components and Weightage

S.No	Credit Structure [L-T-P-C]	Percentage / Marks	CA		Mid-Term		End-term		Project	Total	Exam Conducted by
			Theory	Practical	Theory	Practical	Theory	Practical			
1	3-0-0-3	Percentage	25%	-	25%	-	50%	-	-	100 %	Mid-Term & End Term by CoE
		Marks	50	-	50	-	100	-	-	200	
2	2-0-2-3	Percentage	12.50%	12.50%	12.50 %	12.50%	25%	25%	-	100 %	Mid-Term & End Term by CoE * Except for full stack courses
		Marks	25	25	25	25	50	50	-	200	
3	1-0-4-3	Percentage	-	25%	10%	40%	5%	20%	-	100 %	Mid-Term & End Term by School
		Marks	-	25	10	40	5	20	-	100	
4	2-0-4-4	Percentage	12.50%	12.50%	10%	15%	20%	30%	-	100 %	*Mid-Term & End Term by CoE
		Marks	25	25	20	30	40	60	-	200	
5	0-0-4-2	Percentage	-	50%	-	-	-	-	50%	100 %	Project evaluated by IC at School level
		Marks	-	50	-	-	-	-	50	100	
6	0-0-2-1	Percentage	-	100%	-	-	-	-	-	100 %	Only CA at School Level
		Marks	-	100	-	-	-	-	-	100	
7	3-0-2-4	Percentage	12.50%	12.50%	15%	10%	30%	20%	-	100 %	Mid-Term & End Term by CoE
		Marks	25	25	30	20	60	40	-	200	
8	2-0-0-2	Percentage	25 %	-	25%	-	50%	-	-	100%	Mid-Term & End Term by CoE
		Marks	50	-	50	-	100	-	-	200	

*CSE3150-Front End Full stack development

CSE3151-Java Full Stack Development

CSE3152-.Net Full Stack development

The exact weightages of Evaluation Components shall be clearly specified in the concerned PRC and respective Course Plan.

Normally, for Practice/Skill based Courses, without a defined credit structure (L–T–P) [NTCC], but with assigned Credits (as defined in Clause 5.2 of the Academic Regulations), the method of evaluation shall be based only on Continuous Assessments. The various components of Continuous Assessments, the distribution of weightage among such components, and the method of evaluation/assessment, shall be as decided and indicated in the Course Plan/PRC. The same shall be approved by the respective DAC.

12.6 Minimum Performance Criteria:

12.6.1 Theory only Course and Lab/Practice Embedded Theory Course

A student shall satisfy the following minimum performance criteria to be eligible to earn the credits towards the concerned Course:

a. A student must obtain a minimum of 30% of the total marks/weightage assigned to the End Term Examinations in the concerned Course.

b. The student must obtain a minimum of 40% of the AGGREGATE of the marks/weightage of the components of Continuous Assessments, Mid Term Examinations and End Term Examinations in the concerned Course.

12.6.2 Lab/Practice only Course and Project Based Courses

The student must obtain a minimum of 40% of the AGGREGATE of the marks/weightage of all assessment components in the concerned Course.

12.6.3 A student who fails to meet the minimum performance criteria listed above in a Course shall be declared as "Fail" and given "F" Grade in the concerned Course. For theory Courses, the student shall have to re-appear in the "Make-Up Examinations" as scheduled by the University in any subsequent semester, or, re-appear in the End Term Examinations of the same Course when it is scheduled at the end of the following Semester or Summer Term, if offered. The marks obtained in the Continuous Assessments (other than the End Term Examination) shall be carried forward and be included in computing the final grade, if the student secures the minimum requirements (as per Clause 8.9.1, 8.9.2 of Academic regulations) in the "Make-Up Examinations" of the concerned Course. Further, the student has an option to re-register for the Course and clear the same in the summer term/ subsequent semester if he/she wishes to do so, provided the Course is offered.

13 Additional clarifications - Rules and Guidelines for Transfer of Credits from MOOC, etc. – Note: These are covered in Academic Regulations

The University allows students to acquire credits from other Indian or foreign institutions and/or Massive Open Online Course (MOOC) platforms, subject to prior approval. These credits may be transferred and counted toward fulfilling the minimum credit requirements for the award of a degree. The process of transfer of credits is governed by the following rules and guidelines:

13.1 The transfer of credits shall be examined and recommended by the Equivalence Committee (Refer Annexure B of Academic regulations) and approved by the Dean - Academics.

13.2 Students may earn credits from other Indian or foreign Universities/Institutions with which the University has an MOU, and that MOU shall have specific provisions, rules and guidelines for transfer of credits. These transferred credits shall be counted towards the minimum credit requirements for the award of the degree.

13.3 Students may earn credits by registering for Online Courses offered by *Study Web of Active Learning by Young and Aspiring Minds* (SWAYAM) and *National Program on Technology Enhanced Learning* (NPTEL), or other such recognized Bodies/ Universities/Institutions as approved by the concerned BOS and Academic Council from time to time. The concerned School/Parent Department shall publish/include the approved list of Courses and the rules and guidelines governing such transfer of credits of the concerned Program from time to time. The Rules and Guidelines for the transfer of credits specifically from the Online Courses conducted by SWAYAM/ NPTEL/ other approved MOOCs are as stated in the following Sub-Clauses:

13.3.1 A student may complete SWAYAM/NPTEL/other approved MOOCs as mentioned in Clause 17.3 (as per Academic regulations) and transfer equivalent credits to partially or fully complete the mandatory credit requirements of Discipline Elective Courses and/or the mandatory credit requirements of Open Elective Courses as prescribed in the concerned Curriculum Structure. However, it is the sole responsibility of the student to complete the mandatory credit requirements of the Discipline Elective Courses and the Open Elective Courses as prescribed by the Curriculum Structure of the concerned Program.

13.3.2 SWAYAM/NPTEL/ other approved MOOCs as mentioned in Clause 17.3 (as per Academic regulations) shall be approved by the concerned Board of Studies and placed (as Annexures) in the concerned PRC.

13.3.3 Parent Departments may release a list of SWAYAM/NPTEL/other approved MOOCs for Pre-Registration as per schedule in the Academic Calendar or through University Notification to this effect.

13.3.4 Students may Pre-Register for the SWAYAM/NPTEL/other approved MOOCs in the respective Departments and register for the same Courses as per the schedule announced by respective Online Course Offering body/institute/ university.

13.3.5 A student shall request for transfer of credits only from such approved Courses as mentioned in Sub Clause 17.3.2 above.

13.3.6 SWAYAM/NPTEL/other approved MOOCs Courses are considered for transfer of credits only if the concerned student has successfully completed the SWAYAM/NPTEL/other approved MOOCs and obtained a certificate of successful/satisfactory completion.

13.3.7 A student who has successfully completed the approved SWAYAM/NPTEL/ other approved MOOCs and wants to avail the provision of transfer of equivalent credits, must submit the original Certificate of Completion, or such similar authorized documents to the HOD concerned, with a written request for the transfer of the equivalent credits. On verification of the Certificates/Documents and approval by the HOD concerned, the Course(s) and equivalent Credits shall forwarded to the COE for processing of results of the concerned Academic Term.

13.3.8 The credit equivalence of the SWAYAM/NPTEL/other approved MOOCs are based on Course durations and/or as recommended by the Course offering body/institute/university. The Credit Equivalence mapped to SWAYAM/ NPTEL approved Courses based on Course durations for transfer of credits is summarised in Table shown below. The Grade will be calculated from the marks received by the Absolute Grading Table 8.11 in the Academic regulations.

Table 2: Durations and Credit Equivalence for Transfer of Credits from SWAYAM-NPTEL/ other approved MOOC Courses

Sl. No.	Course Duration	Credit Equivalence
1	4 Weeks	1 Credit
2	8 Weeks	2 Credits
3	12 Weeks	3 Credits

13.3.9 The maximum permissible number of credits that a student may request for credit transfer from MOOCs shall not exceed 20% of the mandatory minimum credit requirements specified by the concerned Program Regulations and Curriculum for the award of the concerned Degree.

13.3.10 The University shall not reimburse any fees/expense; a student may incur for the SWAYAM/NPTEL/other approved MOOCs.

13.4 The maximum number of credits that can be transferred by a student shall be limited to forty percent (40%) of the mandatory minimum credit requirements specified by the concerned Program Regulations and Curriculum for the award of the concerned Degree. However, the grades obtained in the Courses transferred from other Institutions/MOOCs, as mentioned in this Section (13.0), shall not be included in the calculation of the CGPA.

PART B – PROGRAM STRUCTURE

14. Structure / Component with Credit Requirements Course Baskets & Minimum Basket wise Credit Requirements

The B.Tech. (Computer Science and Engineering) Program Structure (2023-2027) totaling 160 credits. Table 3.0 summarizes the type of baskets, number of courses under each basket and the associated credits that are mandatorily required for the completion of the Degree.

Table 3.0: B.Tech. (Computer Science & Engineering) 2023-2027: Summary of Mandatory Courses and Minimum Credit Contribution from various Baskets

Sl. No.	Baskets	Credit Contribution
1	School Core	67
2	Program Core	60
3	Discipline Elective	27
4	Open Elective	6
	Total Credits	**160 (Minimum)**

In the entire Program, the practical and skill based course component contribute to an extent of approximately 64% out of the total credits of 160 for B.Tech. (Computer Science and Engineering) program of four years' duration.

15. Minimum Total Credit Requirements of Award of Degree

As per the AICTE guidelines, a minimum of 160 credits is required for the award of a B.Tech. degree.

16. Other Specific Requirements for Award of Degree, if any, as prescribed by the Statutory Bodies,

16.1 The award of the Degree shall be recommended by the Board of Examinations and approved by the Academic Council and Board of Management of the University**.**

16.2 A student shall be declared to be eligible for the award of the concerned Degree if she/he:

a. Fulfilled the Minimum Credit Requirements and the Minimum Credits requirements under various baskets;

b. Secure a minimum CGPA of 4.50 in the concerned Program at the end of the Semester/Academic Term in which she/he completes all the requirements for the award of the Degree as specified in Sub-Clause 19.2.1 of Academic Regulations**;**

c. No dues to the University, Departments, Hostels, Library, and any other such Centers/ Departments of the University; and

d. No disciplinary action is pending against her/him.

PART C – CURRICULUM STRUCTURE

17. Curriculum Structure – Basket Wise Course List (not Semester Wise)
List of Courses Tabled – aligned to the Program Structure
(Course Code, Course Name, Credit Structure (LTPC), Contact Hours, Course Basket, Type of Skills etc., as applicable).

Table 3.1: List of School core Courses

S.No	Course Code	Course Name	L	T	P	C
1	ENG1002	Technical English	1	0	2	2
2	PPS1001	Introduction to soft skills	0	0	2	1
3	PPS1011	Introduction to Verbal Ability	0	1	0	0
4	ENG2001	Advanced English	1	0	2	2
5	PPS1002	Soft Skills for Engineers	0	0	2	1
6	PPS4002	Introduction to Aptitude	0	0	2	1
7	PPS4004	Aptitude Training Intermediate	0	0	2	1
8	MAT1001	Calculus and Linear Algebra	3	0	2	4
9	PHY1002	Optoelectronics and Device Physics	2	0	2	3
10	MAT1003	Applied Statistics	1	0	2	2
11	MAT1002	Transform Techniques, Partial Differential Equations and Their Applications	3	0	0	3
12	MAT2003	Numerical Methods for Engineers	1	0	2	2
13	MAT2004	Discrete Mathematical Structures	3	0	0	3
14	ECE1001	Elements of Electronics Engineering	3	0	2	4
15	CSE1004	Problem Solving Using C	1	0	4	3
16	CIV1008	Basic Engineering Sciences	2	0	0	2
17	MEC1006	Engineering Graphics	2	0	0	2
18	CSE1006	Problem Solving using JAVA	1	0	4	3
19	ECE2010	Innovative Projects Using Arduino	-	-	-	1
20	ECE2011	Innovative Projects Using Raspberry Pi	-	-	-	1
21	CSE1005	Programming in Python	1	0	4	3
22	CSE2001	Data Structures and Algorithms	3	0	2	4
23	CSE3216	Mastering Object-Oriented Concepts in Python	0	0	2	1
24	CSE2510	Competitive Programming and Problem Solving	0	0	4	2
25	CSE7000	Internship				2
26	CSE7100	Mini Project				4
27	CSE7300	Capstone Project	-	-	-	10
					Total No. of Credits	**67**

Table 3.2: List of Program Core Courses						
S.No	Course Code	Course Name	L	T	P	C
1	ECE2007	Digital Design	2	0	2	3
2	CSE3190	Fundamentals of Data Analytics	2	0	2	3
3	CSE2014	Software Engineering	3	0	0	3
4	CSE2066	Computer Graphics	3	0	0	3
5	CSE3156	Database Management Systems	3	0	2	4
6	CSE2007	Design and Analysis of Algorithms	3	0	0	3
7	CSE2009	Computer Organization and Architecture	3	0	0	3
8	CSE3155	Data Communication and Computer Networks	3	0	2	4
9	CSE3146	Advanced Java Programming	1	0	4	3
10	CSE1700	Essentials of AI	3	0	0	3
11	CSE2508	Mobile Application Development	2	0	0	2
12	CSE2500	Theory of Computation	3	0	0	3
13	CSE2513	Object Oriented Analysis and Design	3	0	0	3
14	CSE2502	Operating Systems	3	0	0	3
15	CSE2511	Operating Systems Lab	0	0	2	1
16	CSE1701	Essentials of AI Lab	0	0	4	2
17	CSE2509	Mobile Application Development Lab	0	0	4	2
18	CSE2506	Cloud Computing	2	0	0	2
19	CSE2514	Compiler Design	3	0	0	3
20	CSE2503	Cryptography and Network Security	3	0	0	3
21	CSE1504	Web Technologies	2	0	0	2
22	CSE2507	Cloud Computing Lab	0	0	2	1
23	CSE1505	Web Technologies Laboratory	0	0	2	1
				Total No. of Credits		**60**

18. Practical / Skill based Courses – Internships / Thesis / Dissertation / Capstone Project Work / Portfolio / Mini project

Practical / Skill based Courses like internship, project work, capstone project, research project / dissertation, and such similar courses, where the pedagogy does not lend itself to a typical L-T-P-C Structure as defined in Clause 5.1 of the Academic Regulations, are simply assigned the number of Credits based on the quantum of work / effort required to fulfill the learning objectives and outcomes prescribed for the concerned Courses. Such courses are referred to as Non-Teaching Credit Courses (NTCC). These Courses are designed to provide students with hands-on experience and skills essential for their professional development. These courses aim to equip students with abilities in problem identification, root cause analysis, problem-solving, innovation, and design thinking through industry exposure and project-based learning. The expected outcomes are first level proficiency in problem solving and design thinking skills to better equip B.Tech. graduates for their professional careers. The method of evaluation and grading for the Practical / Skill based Courses shall be prescribed and approved by the concerned Departmental Academic Committee (refer Annexure A of the Academic Regulations). The same shall be prescribed in the Course Handout.

18.1 Internship

A student may undergo an Internship for a period of 6-8 weeks in an industry / company or academic / research institution during the Semester Break between 4th and 5th Semesters or 6th and 7th Semesters, subject to the following conditions:

18.1.1 The Internship shall be in conducted in accordance with the Internship Policy prescribed by the University from time to time.

18.1.2 The selection criteria (minimum CGPA, pass in all Courses as on date, and any other qualifying criteria) as applicable / stipulated by the concerned Industry / Company or academic / research institution for award of the Internship to a student;

18.1.3 The number of Internships available for the concerned Academic Term. Further, the available number of internships shall be awarded to the students by the University on the basis of merit using the CGPA secured by the student. Provided further, the student fulfils the criteria, as applicable, specified by the Industry / Company or academic / research institution providing the Internship, as stated in Sub-Clause 18.1.2 above.

18.1.4 A student may opt for Internship in an Industry / Company or academic / research institution of her / his choice, subject to the condition that the concerned student takes the responsibility to arrange the Internship on her / his own. Provided further, that the Industry / Company or academic / research institution offering such Internship confirms to the University that the Internship shall be conducted in accordance with the Program Regulations and Internship Policy of the University.

18.1.5 A student selected for an Internship in an industry / company or academic / research institution shall adhere to all the rules and guidelines prescribed in the Internship Policy of the University.

18.2 Project Work

A student may opt to do a Project Work for a period of 6-8 weeks in an Industry / Company or academic / research institution or the University Department(s) as an equivalence of Internship during the Semester Break between 4th and 5th Semesters or 6th and 7th Semesters or during the 5th / 6th / 7th Semester as applicable, subject to the following conditions:

18.2.1 The Project Work shall be approved by the concerned HOD and be carried out under the guidance of a faculty member.

18.2.2 The student may do the project work in an Industry / Company or academic / research institution of her / his choice subject to the above mentioned condition (Sub-Clause 2.6.2.1). Provided further, that the Industry / Company or academic / research institution offering such project work confirms to the University that the project work will be conducted in accordance with the Program Regulations and requirements of the University.

18.3 Capstone Project

A student may undergo a Capstone Project for a period of 10-12 weeks in an industry / company or academic / research institution in the 7th / 8th Semester as applicable, subject to the following conditions:

18.3.1 The Capstone Project shall be in conducted in accordance with the Capstone Project Policy prescribed by the University from time to time.

18.3.2 The selection criteria (minimum CGPA, pass in all Courses as on date, and any other qualifying criteria) as applicable / stipulated by the concerned Industry / Company or academic / research institution for award of the Capstone Project to a student;

18.3.3 The number of Capstone Project available for the concerned Academic Term. Further, the available number of Capstone Project shall be awarded to the students by the University on the basis of merit using the CGPA secured by the student. Provided further, the student fulfils the criteria, as applicable, specified by the Industry / Company or academic / research institution providing the Capstone Project, as stated in Sub-Clause 18.3.2 above.

18.3.4 A student may opt for Capstone Project in an Industry / Company or academic / research institution of her / his choice, subject to the condition that the concerned student takes the responsibility to arrange the I Capstone Project on her / his own. Provided further, that the Industry / Company or academic / research institution offering such Capstone Project confirms to the University that the Capstone Project shall be conducted in accordance with the Program Regulations and Capstone project Policy of the University.

18.3.5 A student selected for a Capstone Project in an industry / company or academic / research institution shall adhere to all the rules and guidelines prescribed in the Capstone Project Policy of the University.

18.4 Research Project / Dissertation

A student may opt to do a Research Project / Dissertation for a period of 12-14 weeks in an Industry / Company or academic / research institution or the University Department(s) as an equivalence of Capstone Project, subject to the following conditions:

18.4.1 The Research Project / Dissertation shall be approved by the concerned HOD and be carried out under the guidance of a faculty member.

The student may do the Research Project / Dissertation in an Industry / Company or academic / research institution of her / his choice subject to the above mentioned condition (Sub-Clause 18.4.1). Provided further, that the Industry / Company or academic / research institution offering such Research Project / Dissertation confirms to the University that the Research Project / Dissertation work will be conducted in accordance with the Program Regulations and requirements of the University.

19. List of Elective Courses under various Specializations / Stream Basket

Table 3.3: Discipline Elective Courses/Specialization Tracks – Minimum of 18 credits is to be earned by the student in a particular track and overall 27 credits.

Track -1 Artificial Intelligence and Machine Learning

Sl.No	Course Code	Course Name	L	T	P	C	Prerequisite
1	CSE3400	Intelligent Systems with Machine Learning	2	0	2	3	CSE1700
2	CSE3401	Advanced Deep Learning Techniques	3	0	0	3	CSE1700
3	CSE3402	Computational Optimization for Intelligent Systems.	3	0	0	3	CSE1700
4	CSE3403	Reinforcement Learning for AI Systems	2	0	2	3	CSE1700
5	CSE3404	Computational Linguistics Natural Language Processing	3	0	0	3	CSE1700
6	CSE3405	Synergistic Neural Fuzzy Computing	2	0	2	3	CSE1700

Track -2 BioInformatics & Data Science

Sl.No	Course Code	Course Name	L	T	P	C	Prerequisite
1	CSE3069	Introduction to Bioinformatics	3	0	0	3	Nil
2	CSE3407	Algorithms in Computational Biology	2	0	2	3	CSE3069
3	CSE3408	Statistical Methods for BioInformatics	2	0	2	3	CSE3069
4	CSE3409	Emerging Technologies in Big Data	2	0	2	3	CSE3156
5	CSE3410	Statistical Techniques of Data Science	2	0	2	3	MAT1003
6	CSE3411	Predictive Analytics and Applications	2	0	2	3	MAT1003
7	CSE2021	Data Mining	3	0	0	3	MAT1003
8	CSE3413	No SQL Data Management	2	0	2	3	CSE3156
9	CSE3414	Applied Data Intelligence	2	0	2	3	Nil

Track -3 Cloud & Security

Sl.No	Course Code	Course Name	L	T	P	C	Prerequisite

1	CSE3415	Cloud Data Engineering	2	0	2	3	CSE3155
2	CSE3416	Federated Learning	2	0	2	3	CSE3155
3	CSE3417	Edge Computing	2	0	2	3	CSE3155
4	CSE3418	Network Security and Firewall Management	2	0	2	3	CSE3155
5	CSE3419	Information Security and Management	3	0	0	3	CSE3155
6	CSE3420	Network Intrusion Detection and Prevention	3	0	0	3	CSE3155
7	CSE3421	Principles and Practices of Web Security	2	0	2	3	CSE3155
8	CSE3422	Penetration Testing and Risk Assessment	3	0	0	3	CSE3155

Track -4 Programming

Sl.No	Course Code	Course Name	L	T	P	C	Prerequisite
1	CSE3423	Go Programming	3	0	0	3	CSE1004
2	CSE3424	Advanced Database Management Systems	2	0	2	3	CSE3156
3	CSE3425	Programming in C# and .NET	1	0	4	3	CSE1006
4	CSE3426	Front End Full Stack Development	2	0	2	3	CSE1006
5	CSE3427	Java Full Stack Development	2	0	2	3	CSE1006
6	CSE3428	.Net Full Stack Development	2	0	2	3	CSE1006
7	CSE3429	Rust Programming	2	0	2	3	CSE1006

Track -5 Fintech and Blockchain

Sl.No	Course Code	Course Name	L	T	P	C	Prerequisite
1	CSE3430	Introduction to Fintech	2	0	2	3	Nil
2	CSE3431	Banking Technology	2	0	2	3	NIl
3	CSE3432	Blockchain Technology	2	0	2	3	Nil
4	CSE3433	Embedded and Decentralized Finance	2	0	2	3	CSE3430
5	CSE3434	Financial and Capital Markets	2	0	2	3	CSE3430
6	CSE3435	Blockchain Development and Programming	2	0	2	3	CSE3432
7	CSE3436	Statistics and Data Analysis for Finance	2	0	2	3	CSE3430
8	CSE3437	Financial Regulations and Compliances	3	0	0	3	CSE3430

Track -6 Special Track

Sl.No	Course Code	Course Name	L	T	P	C	Prerequisite
1	CAI3427	Language Models for Text Mining	2	0	2	3	CSE1700
2	CAI3428	Practical Deep Learning with TensorFlow	2	0	2	3	CSE1700
3	CAI3429	Deep Learning for Computer Vision	2	0	2	3	MAT1003

20. List of Open Electives to be offered by the School / Department (Separately for ODD and EVEN Semesters.

Table 3.4: Open Elective Courses Baskets: Minimum Credits to be earned from this Basket is 6												
Sl. No.	Course Code	Course Name	L	T	P	C	Type of Skill/ Focus	Course Caters to	Prerequisites/ Corequisites	Antirequisites	Future Courses that need this as a Prerequisite	Course Category
Chemistry Basket												
1	CHE1003	Fundamentals of Sensors	3	0	0	3	SD	ES	-	-		OEC
2	CHE1004	Smart materials for IOT	3	0	0	3	SD	ES	-	-		OEC
3	CHE1006	Introduction to Nano technology	3	0	0	3	SD	ES	-	-		OEC
4	CHE1011	Chemical and Petrochemical catalysts	3	0	0	3	SD	ES	-	-		OEC
5	CHE1013	Chemistry for Engineers	3	0	0	3	SD	ES	-	-		OEC
6	CHE1014	Surface and Coatings technology	3	0	0	3	SD	ES	-	-		OEC
7	CHE1016	Forensic Science	3	0	0	3	SD	HP	-	-		OEC
Civil Engineering Basket												
1	CIV1001	Disaster mitigation and management	3	0	0	3	SD	ES / HP	-	-		OEC
2	CIV1002	Environmental Science and Disaster Management	3	0	0	3	FC	ES	-	-		OEC
3	CIV2001	Sustainablility Concepts in Engineering	3	0	0	3	SD	ES	-	-		OEC
4	CIV2002	Occupational Health and Safety	3	0	0	3	SD		-	-		OEC
5	CIV2003	Sustainable Materials and Green Buildings	3	0	0	3	SD / EM	ES	-	-		OEC
6	CIV2004	Integrated Project Management	3	0	0	3	SD / EM / EN	HP / GS	-	-		OEC
7	CIV2005	Environmental Impact Assessment	3	0	0	3	EM / EN	ES	-	-		OEC

8	CIV2006	Infrastructure Systems for Smart Cities	3	0	0	3	EM / EN	ES	-	-		OEC
9	CIV2044	Geospatial Applications for Engineers	2	0	2	3	SD / EM	ES	-	-		OEC
10	CIV2045	Environmental Meteorology	3	0	0	3	SD	ES	-	-		OEC
11	CIV3046	Project Problem Based Learning	3	0	0	3	SD	ES	-	-		OEC
12	CIV3059	Sustainability for Professional Practice	3	0	0	3	EN	ES	-	-		OEC
Commerce Basket												
1	COM2007	Basics of Accounting	3	0	0	3	FC	-	-	-		OEC
Computer Science Basket												
1	CSE1004	Problem Solving Using C	1	0	4	3						OEC
2	CSE1006	Problem Solving Using Java	1	0	4	3						OEC
3	CSE2003	Social Network Analytics	3	0	0	3	SD	GS	-	-		OEC
4	CSE2005	Web design fundamentals	2	0	2	3	SD / EM / EN	-	-	-		OEC
5	CSE3111	Artificial Intelligence : Search Methods For Problem Solving	3	0	0	3	SD / EM / EN	-	-	-		OEC
6	CSE3112	Privacy And Security In Online Social Media	3	0	0	3	SD / EM / EN	-	-	-		OEC
7	CSE3113	Computational Complexity	3	0	0	3	SD / EM / EN	-	-	-		OEC
8	CSE3114	Deep Learning for Computer Vision	3	0	0	3	SD / EM / EN	-	-	-		OEC
9	CSE3115	Learning Analytics Tools	3	0	0	3	SD / EM / EN	-	-	-		OEC
Design Basket												
1	DES2001	Design Thinking	3	0	0	3	SD		-	-		OEC

2	DES2080	Art of Design Language	3	0	0	3	SD		-	-		OEC
3	DES2081	Brand Building in Design	3	0	0	3	SD		-	-		OEC
4	DES2085	Web Design Techniques	3	0	0	3	SD		-	-		OEC
5	DES2089	3D Modeling for Professionals	1	0	4	3	SD		-	-		OEC
6	DES2090	Creative Thinking for Professionals	3	0	0	3	SD		-	-		OEC
7	DES2091	Idea Formulation	3	0	0	3	SD		-	-		OEC
Electrical and Electronics Engineering Basket												
1	EEE1002	IoT based Smart Building Technology	3	0	0	3	SD	-	-	-		OEC
2	EEE1003	Basic Circuit Analysis	3	0	0	3	SD	-	-	-		OEC
3	EEE1004	Fundamentals of Industrial Automation	3	0	0	3	SD	-	-	-		OEC
4	EEE1005	Electric Vehicles & Battery technology	3	0	0	3	SD	-	-	-		OEC
5	EEE1006	Smart Sensors for Engineering Applications	3	0	0	3	SD	-	-	-		OEC
Electronics and Communication Engineering Basket												
1	ECE1003	Fundamentals of Electronics	3	0	0	3	FC	-	-	-		OEC
2	ECE1004	Microprocessor based systems	3	0	0	3	FC	-	-	-		OEC
3	ECE3089	Artificial Neural Networks	3	0	0	3	SD	-	-	-		OEC
4	ECE3097	Smart Electronics in Agriculture	3	0	0	3	FC / EM	-	-	-		OEC
5	ECE3098	Environment Monitoring Systems	3	0	0	3	FC / EM	-	-	-		OEC
6	ECE3102	Consumer Electronics	3	0	0	3	FC / EM	-	-	-		OEC
7	ECE3103	Product Design of Electronic Equipment	3	0	0	3	SD / FC / EM / EN	-	-	-		OEC

8	ECE3106	Introduction to Data Analytics	3	0	0	3	FC / EM	-	-	-		OEC
9	ECE3107	Machine Vision for Robotics	3	0	0	3	FC / EM	-	-	-		OEC
English Basket												
1	ENG1009	Reading Advertisement	3	0	0	3	SD		-	-		OEC
2	ENG1010	Verbal Aptitude for Placement	2	0	2	3	SD		-	-		OEC
3	ENG1011	English for Career Development	3	0	0	3	SD		-	-		OEC
4	ENG1013	Indian English Drama	3	0	0	3			-	-		OEC
5	ENG1014	Logic and Art of Negotiation	2	0	2	3						OEC
Kannada Basket												
1	KAN1003	Kannada Kaipidi	3	0	0	3	SD	-	-	-		OEC
2	KAN2005	Anuvadha Kala Sahithya	3	0	0	3	SD	-	-	-		OEC
3	KAN2006	Vichara Manthana	3	0	0	3	SD	-	-	-		OEC
4	KAN2007	Katha Sahithya Sampada	3	0	0	3	SD	-	-	-		OEC
5	KAN2008	Ranga Pradarshana Kala	3	0	0	3	SD	-	-	-		OEC
Foreign Language Basket												
1	FRL1009	Mandarin Chinese for Beginners	3	0	0	3	SD	-	-	-		OEC
Law Basket												
1	LAW1001	Introduction to Sociology	2	0	0	0	2	F	HP	-	-	OEC
2	LAW2001	Indian Heritage and Culture	2	0	0	0	2	F	HP/GS	-	-	OEC
3	LAW2002	Introdcution to Law of Succession	2	0	0	0	2	F	HP/GS	-	-	OEC
4	LAW2003	Introduction to Company Law	2	0	0	0	2	F	HP	-	-	OEC
5	LAW2004	Introduction to Contracts	2	0	0	2	F	HP	-	-	-	OEC
6	LAW2005	Introduction to Copy Rights Law	2	0	0	2	F	HP	-	-	-	OEC
7	LAW2006	Introduction to Criminal Law	2	0	0	2	F	HP	-	-	-	OEC
8	LAW2007	Introduction to Insurance Law	2	0	0	2	F	HP	-	-	-	OEC
9	LAW2008	Introduction to Labour Law	2	0	0	2	F	HP	-	-	-	OEC

10	LAW2009	Introduction to Law of Marriages	2	0	0	2	F	HP/GS	-	-	-	OEC
11	LAW2010	Introduction to Patent Law	2	0	0	2	F	HP	-	-	-	OEC
12	LAW2011	Introduction to Personal Income Tax	2	0	0	2	F	HP	-	-	-	OEC
13	LAW2012	Introduction to Real Estate Law	2	0	0	2	F	HP	-	-	-	OEC
14	LAW2013	Introduction to Trademark Law	2	0	0	2	F	HP	-	-	-	OEC
15	LAW2014	Introduction to Competition Law	3	0	0	3	F	HP	-	-	-	OEC
16	LAW2015	Cyber Law	3	0	0	3	F	HP	-	-	-	OEC
17	LAW2016	Law on Sexual Harrassment	2	0	0	2	F	HP/GS	-	-	-	OEC
18	LAW2017	Media Laws and Ethics	2	0	0	2	F	HP/GS	-	-	-	OEC
Mathematics Basket												
1	MAT2008	Mathematical Reasoning	3	0	0	3	SD	-	-			OEC
2	MAT2014	Advanced Business Mathematics	3	0	0	3	SD	-	-	-		OEC
3	MAT2041	Functions of Complex Variables	3	0	0	3	SD	-	-	-		OEC
4	MAT2042	Probability and Random Processes	3	0	0	3	SD	-	-	-		OEC
5	MAT2043	Elements of Number Theory	3	0	0	3	SD	-	-	-		OEC
6	MAT2044	Mathematical Modelling and Applications	3	0	0	3	SD	-	-	-		OEC
Mechanical Engineering Basket												
1	MEC1001	Fundamentals of Automobile Engineering	3	0	0	3	SD	-	-	-		OEC
2	MEC1002	Introduction to Matlab and Simulink	3	0	0	3	SD / EM	-	-	-		OEC
3	MEC1003	Engineering Drawing	1	0	4	3	SD	-	-	-		OEC
4	MEC2001	Renewable Energy Systems	3	0	0	3	FC	ES	-	-		OEC
5	MEC2002	Operations Research & Management	3	0	0	3	FC	-	-	-		OEC
6	MEC2003	Supply Chain Management	3	0	0	3	SD / EM / EN	-	-	-		OEC

7	MEC2004	Six Sigma for Professionals	3	0	0	3	SD / EM	-	MEC2008	-		OEC
8	MEC2005	Fundamentals of Aerospace Engineering	3	0	0	3	FC	-	-	-		OEC
9	MEC2006	Safety Engineering	3	0	0	3	SD / EM	ES	-	-		OEC
10	MEC2007	Additive Manufacturing	3	0	0	3	FC / EM	-	-	-		OEC
11	MEC3069	Engineering Optimisation	3	0	0	3	SD / EM	-	-	-		OEC
12	MEC3070	Electronics Waste Management	3	0	0	3	FC / SD	ES	-	-		OEC
13	MEC3071	Hybrid Electric Vehicle Design	3	0	0	3	SD / EM	ES	-	-		OEC
14	MEC3072	Thermal Management of Electronic Appliances	3	0	0	3	SD / EM	-	-	-		OEC
15	MEC3200	Sustainable Technologies and Practices	3	0	0	3	SD / EM	-	-	-		OEC
16	MEC3201	Industry 4.0	3	0	0	3	SD / EM	-	-	-		OEC
Petroleum Engineering Basket												
1	PET1011	Energy Industry Dynamics	3	0	0	3	FC / SD / EM	ES	-	-	-	OEC
2	PET1012	Energy Sustainability Practices	3	0	0	3	FC / SD / EM	ES	-	-	-	OEC
Physics Basket												
1	PHY1003	Mechanics and Physics of Materials	3	0	0	3	FC / SD		-	-		OEC
2	PHY1004	Astronomy	3	0	0	3	FC		-	-		OEC
3	PHY1005	Game Physics	2	0	2	3	FC / SD		-	-		OEC
4	PHY1007	Physics of Nanomaterials	3	0	0	3	FC		-	-		OEC
5	PHY2004	Laser Physics	3	0	0	3	FC	ES	-	-		OEC
6	PHY2005	Science and Technology of Energy	3	0	0	3	FC	ES	-	-		OEC
Management Basket												
1	MGT1001	Introduction to Psychology	3	0	0	3	FC	HP	-	-		OEC
2	MGT1002	Business Intelligence	3	0	0	3	EN		-	-		OEC

3	MGT1003	NGO Management	3	0	0	3	SD		-	-		OEC
4	MGT1004	Essentials of Leadership	3	0	0	3	EM / EN	GS / HP	-	-		OEC
5	MGT1005	Cross Cultural Communication	3	0	0	3	SD / EM / EN	HP	-	-		OEC
6	MGT2001	Business Analytics	3	0	0	3	SD / EM / EN		-	-		OEC
7	MGT2002	Organizational Behaviour	3	0	0	3	FC	HP	-	-		OEC
8	MGT2003	Competitive Intelligence	3	0	0	3	SD	-	-	-		OEC
9	MGT2004	Development of Enterprises	3	0	0	3	SD / EM / EN		-	-		OEC
10	MGT2005	Economics and Cost Estimation	3	0	0	3	SD / EM		-	-		OEC
11	MGT2006	Decision Making Under Uncertainty	3	0	0	3	SD		-	-		OEC
12	MGT2008	Econometrics for Managers	3	0	0	3	SD		-	-		OEC
13	MGT2009	Management Consulting	3	0	0	3	SD / EM / EN		-	-		OEC
14	MGT2010	Managing People and Performance	3	0	0	3	SD / EM / EN	HP / GS	-	-		OEC
15	MGT2011	Personal Finance	3	0	0	3	FC		-	-		OEC
16	MGT2012	E Business for Management	3	0	0	3	SD / EM		-	-		OEC
17	MGT2013	Project Management	3	0	0	3	EN / EM	GS / HP / ES	-	-		OEC
18	MGT2014	Project Finance	3	0	0	3	EN / EM	HP	-	-		OEC
19	MGT2015	Engineering Economics	3	0	0	3	SD		-	-		OEC
20	MGT2016	Business of Entertainment	3	0	0	3	EM / EN		-	-		OEC
21	MGT2017	Principles of Management	3	0	0	3	SD / EM / EN		-	-		OEC
22	MGT2018	Professional and Business Ethics	3	0	0	3	SD / EM / EN	HP	-	-		OEC

23	MGT2019	Sales Techniques	3	0	0	3	SD / EM / EN	HP	-	-		OEC
24	MGT2020	Marketing for Engineers	3	0	0	3	SD / EM / EN	HP				OEC
25	MGT2021	Finance for Engineers	3	0	0	3	SD / EM / EN	HP				OEC
26	MGT2022	Customer Relationship Management	3	0	0	3	SD / EM / EN	HP				OEC
27	MGT2023	People Management	3	0	0	3	SD / EM / EN	HP				OEC
Media Studies Basket												
1	BAJ3051	Digital Photography	2	0	2	3	EM	HP				OEC

21. List of MOOC (NPTEL) Courses

21.1 NPTEL - Discipline Elective Courses for B. Tech. (Computer Science Engineering)

Sl. No.	Course ID	Course Name	Duration
1	noc25-cs22	Deep Learning for Natural Language Processing	12 Weeks
2	noc25-cs49	Machine Learning for Engineering and Science Applications	12 Weeks
3	noc25-cs06	Algorithms in Compuatational Biology and Sequence Analysis	12 Weeks
4	noc25-cs45	Introduction to Large Language Models (LLMs)	12 Weeks
5	noc25-cs61	Quantum Algorithms and Cryptography	12 Weeks

21.2 NPTEL - Open Elective Courses for B. Tech. (Computer Science and Engineering)

Sl. No	Course Code	Course Name	Total Credits	L-T-P-C
1	CSE3111	Artificial Intelligence : Search Methods For Problem Solving	3	3-0-0-3
2	CSE3112	Privacy And Security In Online Social Media	3	3-0-0-3
3	CSE3113	Computational Complexity	3	3-0-0-3
4	CSE3114	Deep Learning for Computer Vision	3	3-0-0-3
5	CSE3115	Learning Analytics Tools	3	3-0-0-3
6	CSE502	Technical Skills in JAVA	3	0-0-6-3
7	CSE503	Technical Skills in Python	3	0-0-6-3
8	CSE504	Comprehensive Technical Skills	5	0-0-10-5
9	CSE505	The Joy Of Computing Using Python	3	3-0-0-3
10	CSE3119	Coding Skills in Python	3	3-0-0-3
11	CSE3121	Parallel Computer Architecture	3	3-0-0-3
12	CSE3124	Games and Information	3	3-0-0-3
13	CSE3140	Introduction To Industry 4.0 And Industrial Internet Of Things	3	3-0-0-3
14	CSE3142	Affective Computing	3	3-0-0-3
15	CSE3112	Privacy and Security in Online Social Media	3	3-0-0-3
16	CSE3196	Foundations of Cyber Physical Systems	3	3-0-0-3
17	CSE3197	Getting Started with Competitive Programming	3	3-0-0-3
18	CSE3198	GPU Architectures And Programming	3	3-0-0-3
19	CSE3199	Artificial Intelligence: Knowledge Representation And Reasoning	3	3-0-0-3
20	CSE3200	Programming in Modern C++	3	3-0-0-3
21	CSE3201	Circuit Complexity Theory	3	3-0-0-3
22	CSE3202	Basics of Computational Complexity	3	3-0-0-3
23	CSE3212	ion to Computer and Network Performance Analysis Using Queuing	1	1-0-0-1
24	CSE3213	C Programming And Assembly Language	1	1-0-0-1
25	CSE3214	Python For Data Science	1	1-0-0-1
26	CSE3215	Software Conceptual Design	1	1-0-0-1
27	CSE3117	Industrial Digital Transformation	3	3-0-0-3
28	CSE3118	Blockchain for Decision Makers	3	3-0-0-3
29	CSE3349	Technology for Lawyers	3	3-0-0-3
30	CSEXXXX	Deep Learning for Natural Language Processing	3	3-0-0-3
31	CSEXXXX	Machine Learning for Engineering and science applications	3	3-0-0-3
32	CSEXXXX	Algorithms in Computational Biology and Sequence Analysis	3	3-0-0-3
33	CSEXXXX	Introduction to Large Language Models (LLMs)	3	3-0-0-3
34	CSEXXXX	Quantum Algorithms and Cryptography	3	3-0-0-3

22. Recommended Semester Wise Course Structure / Flow including the Programme / Discipline Elective Paths / Options

First year - CYCLE 1

Semester 1 - Physics Cycle								
S. NO.	COURSE CODE	COURSE NAME	CREDIT STRUCTURE					BASKET
			L	T	P	C	CONTACT HOURS	
1	MAT1001	Calculus and Linear Algebra	3	0	2	4	5	School Core
2	PHY1002	Optoelectronics and Device Physics	2	0	2	3	4	School Core
3	ECE1001	Elements of Electronics Engineering	3	0	2	4	5	School Core
4	ENG1002	Technical English	1	0	2	2	3	School Core
5	PPS1001	Introduction to soft skills	0	0	2	1	2	School Core
6	CSE1004	Problem Solving Using C	1	0	4	3	5	School Core
7	CHE1018	Environmental Science	1	0	2	0	3	School Core
8	PPS1011	Introduction to Verbal Ability	0	1	0	0	1	School Core
		Total	11	1	16	17	28	

Semester 2 - Engineering Science Cycle								
S. NO.	COURSE CODE	COURSE NAME	CREDIT STRUCTURE					BASKET
			L	T	P	C	CONTACT HOURS	
1	MAT1003	Applied Statistics	1	0	2	2	3	School Core
2	ECE2007	Digital Design	2	0	2	3	4	Program Core
3	CIV1008	Basic Engineering Sciences	2	0	0	2	2	School Core
4	MEC1006	Engineering Graphics	2	0	0	2	2	School Core
5	CSE1006	Problem Solving using JAVA	1	0	4	3	5	School Core
6	ENG2001	Advanced English	1	0	2	2	3	School Core
7	PPS1002	Soft Skills for Engineers	0	0	2	1	2	School Core
8	ECE2010	Innovative Projects Using Arduino	-	-	-	1	-	School Core
		Total	9	0	12	16	21	

First year - CYCLE 2

Semester 1 - Engineering Science Cycle								
S. NO.	COURSE CODE	COURSE NAME	CREDIT STRUCTURE					BASKET
			L	T	P	C	CONTACT HOURS	
1	MAT1001	Calculus and Linear Algebra	3	0	2	4	5	School Core
2	ECE1001	Elements of Electronics Engineering	3	0	2	4	5	School Core
3	ENG1002	Technical English	1	0	2	2	3	School Core
4	PPS1001	Introduction to soft skills	0	0	2	1	2	School Core
5	CSE1004	Problem Solving Using C	1	0	4	3	5	School Core
6	PPS1011	Introduction to Verbal Ability	0	1	0	0	1	School Core
7	CIV1008	Basic Engineering Sciences	2	0	0	2	2	School Core
8	MEC1006	Engineering Graphics	2	0	0	2	2	School Core
		Total	12	1	12	18	25	

Semester 2 - Physics Cycle								
S. NO.	COURSE CODE	COURSE NAME	CREDIT STRUCTURE					BASKET
			L	T	P	C	CONTACT HOURS	
1	MAT1003	Applied Statistics	1	0	2	2	3	School Core
2	ECE2007	Digital Design	2	0	2	3	4	Program Core
3	CSE1006	Problem Solving using JAVA	1	0	4	3	5	School Core
4	ENG2001	Advanced English	1	0	2	2	3	School Core
5	PPS1002	Soft Skills for Engineers	0	0	2	1	2	School Core
6	CHE1018	Environmental Science	1	0	2	0	3	School Core
7	PHY1002	Optoelectronics and Device Physics	2	0	2	3	4	School Core
8	ECE2010	Innovative Projects Using Arduino	-	-	-	1	-	School Core
		Total	8	0	16	15	24	

Semester 3								
S. NO.	COURSE CODE	COURSE NAME	CREDIT STRUCTURE					BASKET
			L	T	P	C	CONTACT HOURS	
1	MAT1002	Transform Techniques, Partial Differential Equations and Their Applications	3	0	0	3	3	School Core

2	ECE2011	Innovative Projects Using Raspberry Pi	-	-	-	1	-	School Core
3	CSE1005	Programming in Python	1	0	4	3	5	School Core
4	CSE3190	Fundamentals of Data Analytics	2	0	2	3	4	Program Core
5	CSE2014	Software Engineering	3	0	0	3	3	Program Core
6	CSE2066	Computer Graphics	3	0	0	3	3	Program Core
7	CSE3156	Database Management Systems	3	0	2	4	5	Program Core
8	PPS4002	Introduction to Aptitude	0	0	2	1	2	School Core
9	CSE2001	Data Structures and Algorithms	3	0	2	4	5	School Core
		Total	18	0	12	25	30	

Semester 4

S. NO.	COURSE CODE	COURSE NAME	CREDIT STRUCTURE					BASKET
			L	T	P	C	CONTACT HOURS	
1	MAT2003	Numerical Methods for Engineers	1	0	2	2	3	School Core
2	CSE2007	Design and Analysis of Algorithms	3	0	0	3	3	Program Core
3	MAT2004	Discrete Mathematical Structures	3	0	0	3	3	School Core
4	CSE2009	Computer Organization and Architecture	3	0	0	3	3	Program Core
5	CSE3155	Data Communication and Computer Networks	3	0	2	4	5	Program Core
6	CSE3146	Advanced Java Programming	1	0	4	3	5	Program Core
7	CSE XXXX	Discipline Elective – I	3	0	0	3	3	Discipline Elective
8	XXX XXXX	Open Elective – I (Management Basket)	3	0	0	3	3	Open Elective
9	PPS4004	Aptitude Training Intermediate	0	0	2	1	2	School Core
10	CSE3216	Mastering Object-Oriented Concepts in Python	0	0	2	1	2	School Core
		Total	20	0	12	26	32	

Semester 5

S. NO.	COURSE CODE	COURSE NAME	CREDIT STRUCTURE					BASKET
			L	T	P	C	CONTACT HOURS	
1	CSE1700	Essentials of AI	3	0	0	3	3	Program Core
2	CSE2508	Mobile Application Development	2	0	0	2	2	Program Core
3	CSE2500	Theory of Computation	3	0	0	3	3	Program Core

4	CSE2513	Object Oriented Analysis and Design	3	0	0	3	3	Program Core
5	CSE2502	Operating Systems	3	0	0	3	3	Program Core
6	CSE XXXX	Discipline Elective - II	3	0	0	3	3	Discipline Elective
7	CSE XXXX	Discipline Elective - III	3	0	0	3	3	Discipline Elective
8	CSE1701	Essentials of AI Lab	0	0	4	2	4	Program Core
9	CSE2509	Mobile Application Development Lab	0	0	4	2	4	Program Core
10	CSE2511	Operating Systems Lab	0	0	2	1	2	Program Core
11	CSE7000	Internship	-	-	-	2	-	School Core
		Total	20	0	10	27	30	

Semester 6								
S. NO.	COURSE CODE	COURSE NAME	CREDIT STRUCTURE					BASKET
			L	T	P	C	CONTACT HOURS	
1	CSE2506	Cloud Computing	2	0	0	2	2	Program Core
2	CSE2514	Compiler Design	3	0	0	3	3	Program Core
3	CSE2503	Cryptography and Network Security	3	0	0	3	3	Program Core
4	CSE1504	Web Technologies	2	0	0	2	2	Program Core
5	CSEXXXX	Discipline Elective - IV	3	0	0	3	3	Discipline Elective
6	CSEXXXX	Discipline Elective - V	3	0	0	3	3	Discipline Elective
7	CSEXXXX	Discipline Elective - VI	3	0	0	3	3	Discipline Elective
8	CSE2510	Competitive Programming and Problem Solving	0	0	4	2	4	School Core
9	CSE2507	Cloud Computing Lab	0	0	2	1	2	Program Core
10	CSE1505	Web Technologies Lab	0	0	2	1	2	Program Core
11	PPSXXXX	Industry Preparedness Program	2	0	0	0	2	School Core
		Total	21	0	8	23	29	

Semester 7								
S. NO.	COURSE CODE	COURSE NAME	CREDIT STRUCTURE					BASKET
			L	T	P	C	CONTACT HOURS	
1	CSE7100	Mini Project				4	-	School Core
2	XXXXXXX	Open Elective II	3	0	0	3	3	Open Elective

3	CSEXXXX	Discipline Elective - VII	3	0	0	3	3	Discipline Elective
4	CSEXXXX	Discipline Elective - VIII	3	0	0	3	3	Discipline Elective
5	CSEXXXX	Discipline Elective - IX	3	0	0	3	3	Discipline Elective
		Total	12	0	0	16	12	

Semester 8								
S. NO.	COURSE CODE	COURSE NAME	CREDIT STRUCTURE					BASKET
			L	T	P	C	CONTACT HOURS	
1	CSE7300	Capstone Project	-	-	-	10	-	School Core
		Total	0	0	0	10	-	

23. Course Catalogue

Course Catalogue of all Courses Listed including the Courses Offered by other School / Department and Discipline / Programme Electives – Course Code, Course Name, Prerequisite, Anti-requisite, Course Description, Course Outcome, Course Content (with Blooms Level, CO, No. of Contact Hours), Reference Resources.

urse Code: AT1001	**urse Title: Calculus and Linear Algebra** **pe of Course: School Core** **o Integrated**	**- P- C**	**2**	**1**	**2**	**4**
rsion No.	**)**					
urse Pre-requisites	sic Concepts of Limits, Differentiation, Integration					
ti-requisites	**.**					
urse Description	The course focuses on the concepts of calculus and linear algebra with reference to specific engineering problems. The course is of both conceptual and analytical type in nature. The lab sessions associated with the course are concerned with acquiring an ability to use the MATLAB software.					
urse Objective	The objective of the course is **Skill Development** of student by using **Problem Solving Techniques.**					
urse Out Comes	On successful completion of the course the students shall be able to: 1) Comprehend the knowledge of applications of matrix principles. 2) Understand the concept of partial derivatives and their applications. 3) Apply the principles of integral calculus to evaluate integrals. 4) Adopt the various analytical methods to solve differential equations. 5) Demonstrate the use of MATLAB software to deal with a variety of mathematical problems.					
urse Content:						
odule 1	**ear Algebra**				**10 Classes**	
Review: Types of matrices, elementary transformations, rank of a matrix, normal form, Solution of systems of linear equations: (Homogenous and non-homogenous system) AX = O and AX = B using rank method. **Linear Algebra:** envalues and Eigenvectors of a real matrix – Characteristic equation – Properties of Eigenvalues and Eigenvectors – Cayley-Hamilton theorem – Diagonalization of matrices – Reduction of a quadratic form to canonical form by orthogonal transformation – Nature of quadratic forms. gineering Applications of Linear Algebra.						
odule 2	**rtial Derivatives**				**10 CLASSES**	
Review: Differential calculus with single variable. **Partial Derivatives:**						

Homogeneous functions and Euler's theorem, Total derivative, Change of variables, Jacobians, Partial differentiation of implicit functions, Taylor's series for functions of two variables, Maxima and minima of functions of two variables, Lagrange's method of undetermined multipliers.

gineering Applications of partial derivatives.

dule 3	**vanced Integral calculus**			**12 Classes**

Review: Integral calculus for single integrals.

Advanced Integral calculus:

Beta and Gamma functions–interrelation-evaluation of integrals using gamma and beta functions; error function-properties. Multiple Integrals- Double integrals – Change of order of integration – Double integrals in polar coordinates – Area enclosed by plane curves, evaluation of triple integrals-change of variables between Cartesian and cylindrical and spherical polar co-ordinates.

gineering applications of partial derivatives.

dule 4	**dinary Differential Equations**	**signment**	**ogramming**	**12 Classes**

Review: First order and first-degree Ordinary Differential Equations, Method of separation of variables, Homogeneous and Non- Homogeneous Equations reducible to Homogeneous form.

Linear Differential Equations, Bernoulli's Differential Equation, Exact and Non- Exact Differential Equations, Higher order Differential Equation with constant coefficients and with right hand side of the form e^{ax}, sinax, cosax, $e^{ax}f(x)$, $x^nf(x)$ etc., Linear equations with variable coefficients such as Cauchy Equation and Lagrange's Equation, D-operators and Inverse D- operators, Method of Variation of Parameters.

gineering applications of differential equations.

t of Laboratory Tasks:

roductory Task: Introduction to usage of the software and simple programming tasks. [3 Sessions]

periment N0 1: Solution of Simple differentiation with single variable and use of chain Rule.

periment No. 2: Solution based on application of Tailors' Series using software

periment No. 3: Application of Maxima and Minima condition using software.

periment No. 4 Computation of different functions for a specific problem

periment No. 5 Computation of Area under a curve.

periment No. 6 Solution of a set of simultaneous equations in matrix method

periment No. 7 Computation of Eigen Values and Eigen Vectors.

periment No. 8 Solution of Partial Differential equation

periment No. 9 solution using Cauchy Equation and Lagrange's Equation
rgeted Application & Tools that can be used: e contents of this course has direct applications in most of the core engineering courses for problem formulations, Problem Solution and system Design. ols Used: MatLab, Zylink.
signment:
1. List at least 3 sets of Matrix Applications concerning the respective branch of Engineering and obtain the solution using MATLAB. 2. Select any one simple differential equation pertaining to the respective branch of engineering, identify the dependent and independent variable – Obtain the solution and compare the solution sets by varying the values of the dependent variable.
xt Book 1. Sankara Rao, Introduction to Partial differential equations, Prentice Hall of India, edition, 2011 2. B. S. Grewal (2017), Higher Engineering Mathematics by, 44th Edition, Khanna Publishers.
ferences: 1. Victor Henner, Tatyana Belozerova, Mickhail Khenner, Ordinary and Partial Differential Equations, CRC Press, Edition, 2013. 2. Walter Ledermann, Multiple integrals, Springer, 1st edition 3. Lay, Linear Algebra ansd its applications, 3rd Ed., 2002, Pearson Education India. 4. Erwin Kreyzig, Advanced Engineering Mathematics, John Wiley and sons, Inc.10th Edition 5. MatLab usage manual **E-resources/ Web links:** 1. https://nptel.ac.in/courses/109104124 2. https://nptel.ac.in/courses/111106051 3. https://nptel.ac.in/courses/111102137 4. https://www.cuemath.com/learn/mathematics/algebra-vs-calculus/ 5. https://stanford.edu/~shervine/teaching/cs-229/refresher-algebra-calculus 6. https://math.hmc.edu/calculus/hmc-mathematics-calculus-online-tutorials/linear-algebra/ 7. https://www.math.hkust.edu.hk/~maqian/ma006_0607F.html 8. https://www.scu.edu.au/study-at-scu/units/math1005/2022/
pics relevant to the development of Foundation Skills: All solution methods **pics relevant to development of Employability skills: Use of Matlab software.**

urse Code: Y1002	**Course Title: Optoelectronics and Device Physics** **pe of Course: 1] School Core & Laboratory integrated**	-P-C	)-2-3
rsion No.	)		
urse Pre-requisites	L		
ti-requisites	L		
urse Description	The purpose of this course is to enable the students to understand the fundamentals, working and applications of optoelectronic devices and to develop the basic abilities to appreciate the applications of advanced microscopy and quantum computers. The course develops the critical thinking, experimental and analytical skills. The associated laboratory provides an opportunity to validate the concepts taught and enhances the ability to use the concepts for technological applications. The laboratory tasks aim to develop following skills: **An attitude of enquiry, confidence and ability to tackle new problems, ability to interpret events and results, observe and measure physical phenomena, select suitable equipment, instrument and materials, locate faults in systems.**		
urse Out Comes	On successful completion of the course the students shall be able to: CO1: Describe the concepts of semiconductors, magnetic materials and superconductors. CO2: Apply the concept of materials in the working of optoelectronic and magnetic devices. CO3: Discuss the quantum concepts used in advanced microscopy and quantum computers. CO4: Explain the applications of lasers and optical fibers in various technological fields. CO5: Interpret the results of various experiments to verify the concepts used in optoelectronics and advanced devices. **[Lab oriented].**		
urse Objective	The objective of the course is to familiarize the learners with the concepts of "Optoelectronics and device physics "and attain **Skill Development** through **Experiential Learning** techniques		
urse Content:			

odule 1	**ndamentals of Materials.**	signment	tting of magnetization (M) v/s Magnetic field (H) for diamagnetic, paramagnetic and ferromagnetic materials using excel/ origin software.	**No. of Classes: 07**
	Topics: Concept of energy bands, charge carriers, carrier concentration, concept of Fermi level, Hall effect, Magnetic materials, Superconductors:			
odule 2	**vanced Devices and applications**	signment	ta collection on efficiency of solar cells.	**No. of Classes: 8**
	Topics: p-n junctions, Zener diode, transistor characteristics, Optoelectronic devices:, Solar cells, I-V characteristics, and LEDs			
odule 3	**antum concepts and Applications**	rm paper	minar on quantum computers.	**No. of classes: 8**
	Topics: Planck's quantum theory, applications of Quantum theory: de-Broglie hypothesis, matter waves, properties. de-Broglie wavelength associated with an electron. Heisenberg's uncertainty principle. Schrodinger time independent wave equation. Particle in a box			
odule 4	**sers and Optical fibers**	rm paper	se study on medical applications of Lasers.	**No. of classes :07**
	Topics: Interactions of radiations with matter, Characteristics of laser, conditions and requisites of laser, Modern day applications of laser: LIDAR, LASIK, Cutting, Welding and Drilling. Principle of optical fibers, Numerical aperture and acceptance angle (Qualitative), Attenuation, Applications: Point to point communication with block diagram, application of optical fibers in endoscopy.			
	t of Laboratory Tasks: Experiment No. 1: Experimental errors and uncertainty using excel Level 1: Calculation of accuracy and precision of a given data vel 2: propagation of errors in addition, subtraction, multiplication and division. Experiment N0 2: To determine the wavelength of semiconductor diode Laser and to estimate the particle size of lycopodium powder using diffraction. vel 1: Determination of Wavelength of Laser vel 2: Finding the particle size of lycopodium powder.			

	periment No. 3: To determine the proportionality of Hall Voltage, magnetic flux density and the polarity of Charge carrier. vel 1: To determine the proportionality of Hall Voltage and magnetic flux density vel 2: To determine the polarity of Charge carrier. Experiment No. 4: To study the I-V characteristics of a given zener diode in forward and reverse bias conditions. vel 1: To study I –V characteristics of the given Zener diode in reverse bias and to determine break down voltage. vel 2: To study I –V characteristics of the given Zener diode in forward bias and to determine knee voltage and forward resistance. Experiment No. 5: To study input and output characteristics of a given Transistor. vel 1: To determine the input resistance of a given transistor. vel 2: To determine current transfer characteristics and transistor parameters of a given transistor. Experiment No. 6: Determination of Fermi energy and Fermi temperature of a given metal and bimetallic wire. Level 1: Determination of Fermi energy and Fermi temperature of given metal wire. Level 2: Determination of Fermi energy and Fermi temperature of given bimetallic wire. Experiment No. 7: To study the current vs voltage characteristics of CdS photo-resistor at constant irradiance and To measure the photo-current as a function of the irradiance at constant voltage. vel 1 To study the current vs voltage characteristics of CdS photo-resistor at constant irradiance. vel 2: To measure the photo-current as a function of the irradiance at constant voltage. periment No. 8: To study the I-V characteristics and I-R characteristics of a solar cell as a function of the irradiance. vel 1: To study the I-V characteristics vel 2: I-R characteristics of a solar cell as a function of the irradiance. periment No. 9: Calculate the numerical aperture and study the losses that occur in optical fiber cable. . vel 1: Calculate the numerical aperture. vel 2: study the losses that occur in optical fiber cable. periment No. 10: To determine the magnetic susceptibility of a given diamagnetic and paramagnetic substances using Quincke's method.

	vel 1: To determine the magnetic susceptibility of a given diamagnetic substance. vel 2: To determine the magnetic susceptibility of a given paramagnetic substance. periment No. 11: Plotting I-V characteristics in forward and reverse bias for LEDs d Determination of knee voltage. vel 1: Plotting I-V characteristics in forward and reverse bias for LEDs vel 2: Determination of knee voltage. periment No. 12: Determination of Stefan's constant and verification of Stefan-Boltzmann Law. vel 1: Determination of Stefan's constant vel 2: Verification of Stefan-Boltzmann Law.
	rgeted Application & Tools that can be used: 1. Areas of application are optoelectronics industry, Solar panel technologies, quantum computing software, electronic devices using transistors and diodes, memory devices, endoscopy, SQUIDS in MRI, Advanced material characterizations using SEM and STM. **2.** Origin, excel and Mat lab soft wares for programming and data analysis.
	oject work/Assignment: Mention the Type of Project /Assignment proposed for this course
	Assessment Type • Midterm exam • Assignment (review of digital/ e-resource from PU link given in references section - mandatory to submit screen shot accessing digital resource.) • Quiz • End Term Exam • Self-Learning 1. Prepare a comprehensive report on non-conventional energy resources in Karnataka and their pros and cons. 2. Write a report on importance of quantum entanglement in supercomputers.
	xt Book 1. Engineering Physics by Avadhanalu, Revised edition, S. Chand Publications,2018.
	References: 1. Elementary Solid state Physics: Principles and Applications by M.A. Omar, 1st Edition, Pearson Publications, 2002. 2. Principles of Quantum Mechanics by R Shankar, 2nd edition, springer

	Publications, 2011. 3. Optoelectronics: An Introduction by John Wilson and John Hawkes, 3rd edition, Pearson Publications, 2017. 4. Engineering Physics by Gaur and Gupta, Dhanpat Rai Publications, 2012. 5. Introduction to Quantum Mechanics, David J Griffiths, Cambridge University Press, 2019
	E-Resourses: 1. https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=553045&site=ehost-live 2. https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=833068&site=ehost-live 3. https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=323988&site=ehost-live 4. https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=1530910&site=ehost-live 5. https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=486032&site=ehost-live
	Topics relevant to "SKILL DEVELOPMENT": Fundamentals of materials, Lasers and optical fibers. for Skill Development through Participative Learning Techniques. This is attained through the Assignment/ Presentation as mentioned in the assessment component in course handout.

Course Code: ECE1001	**Course Title: Elements of Electronics Engineering** **Type of Course: School Core Theory & Integrated Laboratory**	**L-T-P-C**	3	0	2	4
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	Nil					
Course Description	The purpose of this course is to enable the students to learn the fundamental concepts of electronic devices and circuits. The course aims at nurturing the students with the fundamental principles of electronics engineering, prevailing in various engineering applications. The nature of the course is conceptual and analytical which imparts knowledge of electronic components and their behavior under various operating conditions. The course develops thinking skills of the students, encouraging their quest for knowledge about electronic devices and their usage in higher semester courses. The associated laboratory provides an opportunity to validate the concepts taught in theory classes and enable the students to work with basic electronic circuits using electronics components.					
Course Objectives	The objective of the course is to familiarize the learners with the concepts of Elements of Electronics Engineering and attain SKILL DEVELOPMENT through EXPERIENTIAL LEARNING .					
Course Outcomes	On successful completion of this course the students shall be able to: tify various electrical and electronic components and basic electrical laws. ainapplications of Diodes and BJTs. marize the concepts of Digital Electronics and Communication Systems. uss the basic concepts of microprocessorand computer organization. orm experiments to familiarizevarious Electrical & Electronic components and equipment. y Basic Electrical Circuit configurations and Laws.					
Course Content:						
Module 1	Basic Electrical and Electronic Components	Assignment / Quiz	Identification of Practical electronic and electrical components / Memory Recall based Quizzes			10 Sessions
Topics: ELECTRICAL CIRCUITS AND LAWS:DC Circuits: Classification of Electrical Elements, Ohm's law, Series and Parallel Circuits, Kirchhoff's Voltage and Current laws, Power and Energy, Transformers and their types. ELECTRONIC MATERIALS AND COMPONENTS: Conductors, Insulators, Semi-Conductor Material, P-N Junction diode, Characteristics and Parameters, Ideal Diode approximations, DC load line.						
Module 2	Applications of Diodes and Introduction to BJT	Assignment / Quiz		Simulation Task/ Memory Recall based Quizzes		12 Sessions
Topics: RECTIFIERS: Half-wave rectifier, Two-diode Full-wave rectifier, Bridge rectifier, Capacitor filter circuit (only qualitative approach). ZENER DIODE: Zener diode, Zener Characteristics, Zener diode as a voltage regulator. BIPOLAR JUNCTION TRANSISTORS: BJT Construction and Operation, BJT Voltages and Currents, Common Base, Common Emitter Configuration and Characteristics, Current amplification Factor alpha and beta, DC Load line w.r.t. fixed bias circuit (Q-Point), AC Analysis.						
Module 3	Digital Electronics and Communication System	Assignment / Quiz		Simulation Task / Memory Recall based Quizzes		13 Sessions

Topics:
NUMBER SYSTEMS: Decimal Number System, Binary Number System, Hexadecimal Number System, Conversions: Binary to and from Hexadecimal; Hexadecimal to and from Decimal;1's and 2's Complement of Binary Numbers, Binary Addition.
BOOLEAN ALGEBRA: Boolean Laws and Theorems, De Morgan's theorem. Digital Circuits: Logic gates, NOT Gate, AND Gate, OR Gate, XOR Gate, X-NOR Gate, NAND Gate, NOR Gate.
COMMUNICATION SYSTEM: Block diagram of communication system, Modulation: Definition of Modulation, Need of Modulation, Types of Modulation: Amplitude Modulation and Frequency Modulation (Waveforms only).

Module 4	Microprocessors and Computer Organization	Assignment / Quiz	Memory recall based Quizzes	10 Sessions

Topics:
INTEL 8085 MICROPROCESSOR: Basic Architecture and features of 8085 Microprocessor.
COMPUTER ORGANISATION:Basic structure of Computer Organisation describing the various Computer types, Functional Units, Basic Operational concepts, Bus Structures, Memory System: RAM and ROM.

List of Laboratory Tasks:

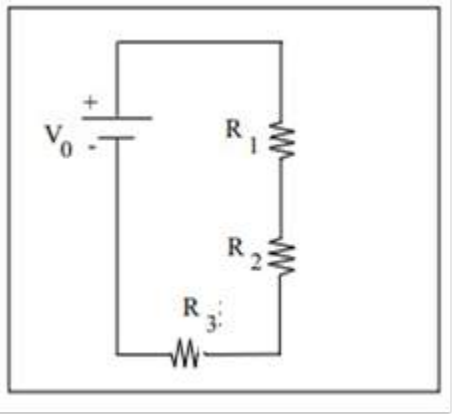
Experiment No. 1:Study of Resistors, Measuring instruments and DC Power Supply.
Level 1:Identification of resistor values from color bands and verification with Multimeter.
Level 2:Connecting a resistive circuit to a DC Power Supply and observing the input and output values using Voltmeters, Ammeters and hence calculate resistance values.

Experiment No. 2:Study of Reactive components, Multimeter, CRO and Function Generator.
Level 1:Identification of various types of capacitive and inductive components and verification with Multimeter.
Level 2:Connecting a reactive circuit to a function generator and observing the input and output waveform on CRO and calculation of Reactance and Impedance.

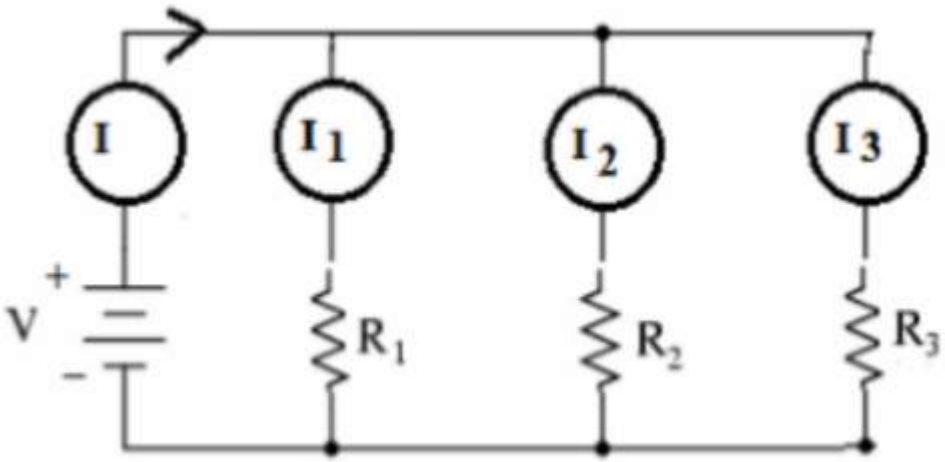
Experiment No. 3: Study of Ohm's Law.
Level 1:Rig up the circuit and verify Ohm's Law.
Level 2: Connect a 100Ω Resistor to a Voltage source of 0-5V. Plot a V- I graph by tabulating the Voltage Vs Current Values accordingly. Repeat the experiment for 1KΩ resistor and compare the results.

Experiment No. 4:Study of Series and Parallel Resistor Connections.
Level 1:Carry out the equivalent resistance of given four resistors 100Ω each connected in series and parallel combination using breadboard.
Level 2:Rig up a Current Divider Circuit and a Voltage Divider Circuit and verify the results.

Experiment No. 5:Study of Kirchhoff's Voltage Law and Kirchhoff's Current Law.
Level 1:Verify KVLand KCL with circuit(a) and circuit(b) with #values.

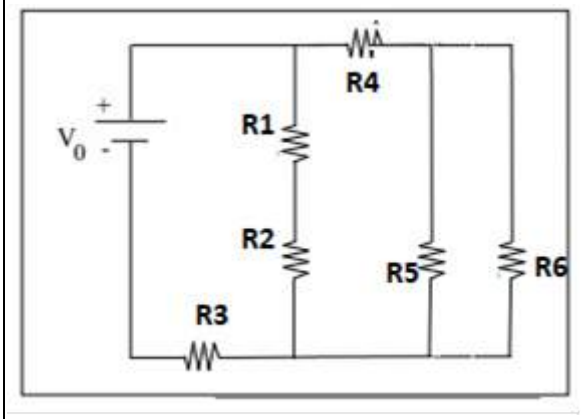


(a)



(b)

Level 2: Verify KCL with the help of given circuit having # values and carry out the equivalent resistance of the circuit by experimental and analytical methods.



Experiment No. 6: Study of PN-Junction Diode Characteristics in Forward and Reverse Bias Conditions.
Level 1:Carry out the experiment to find cut-in voltage on forward characteristics for Silicon P-N Junction diode.
Level 2: Carry out experiment to plot VI Characteristics of Silicon P-N Junction Diode in both forward and reverse biased conditions for Si P-N Junction diode.

Experiment No. 7: Study of Bipolar Junction Transistor in different regions of operation.
Level 1:Carry out the experiment to understand the importance of active, cut off and saturation regions.
Level 2: Carry out the experiment to design and analyze the operation of transistor as switch.

Experiment No. 8: Study of basic Digital Logic Gates using Integrated Chips IC's: NOT, AND, OR, XOR, NAND and NOR Gates
Level 1:Carry out the experiment to study and verify the truth table of logic gates using Digital ICs.
Level 2:Implementation of operation of a basic Boolean expression using basic gates.

Experiment No. 9: Study of Computer Organization: Identification of Components on Motherboard: CPU: Processor Chips (Processor Socket), PCI, Parallel Ports, Universal Serial Bus: USB, I/O Connectors, RAM Slots.
Level 1:Carry out the experiment to familiarize a computer system layout and mark the positions of SMPS, Motherboard, FDD, HDD, CD / DVD drive and add on cards.
Level 2:Study of a Desktop PC and its assembling.

Targeted Application & Tools that can be used:
Student will be able to find career opportunities in various domains such asAnalog Electronics, Digital Electronics, Microprocessors, VLSI Design, Telecommunication, Computers andWireless Communication. The students will be able to join a profession which involves basics to high level of electronic circuit design.

Professionally Used Software: MultiSim/ PSpice
Besides these software tools hardware equipment such as Multimeters, Function Generators, Power Supplies, Oscilloscopes etc., can be used to perform component/circuit testing and analysis.

Textbook(s):
T1. John Hiley, Keith Brown and Ian McKenzie Smith, *"Hughes Electrical and Electronic Technology",* Pearson,12th Edition
T2.William Stallings,"Computer Organization and Architecture Designing for Performance", Pearson Education, 10th Edition.

Reference(s):
Reference Book(s):
R1. Smarajit Ghosh, *"Fundamentals of Electrical and Electronics Engineering*", PHI, 2nd Edition
R2. D.P. Kothari, I. J. Nagrath, *"Basic Electronics",* McGraw Hill Education, 1st Edition
R3. Rajendra Prasad, *"Fundamentals of Electronics Engineering",* Cengane Learning, 3rd Edition

Online Resources (e-books, notes, ppts, video lectures etc.):

Video lectures on "BASIC ELECTRONICS" by Prof. Dr. Chitralekha Mahanta, Department of Electronics and communication Engineering, IIT Guwahati": https://nptel.ac.in/courses/117/103/117103063/

Lecture Series on " Useful Laws in Basic Electronics" by Prof. T.S.Natarajan, Department of physics, IIT Madras: https://www.youtube.com/watch?v=vfVVF58FtCc

Lecture Series on "Introduction to Bipolar Junction Transistors BJT " by All About Electronics Youtube Channel: https://www.youtube.com/watch?v=-VwPSDQmdjM&list=PLwjK_iyK4LLDoFG8FeiKAr3IStRkPSxqq

Lecture Series on " PN Junction Diode " by All About Electronics Youtube Channel: https://www.youtube.com/watch?v=USrY0JspDEg

Lecture Series on "Introduction to Digital Electronics" by All About Electronics Youtube Channel: https://www.youtube.com/watch?v=DBTna2ydmC0&list=PLwjK_iyK4LLBC_so3odA64E2MLgIRKafl

Lecture Series on "Introduction to Microprocessors" by Bharat Acharya Education :https://www.youtube.com/watch?v=0M74z5jEAyA

Lecture Notes on : "Electronic Devices", Bipolar Junction Transistors, 2nd Chapter, by Shree Krishna Khadka (PDF) Bipolar Junction Transistor (researchgate.net)https://www.researchgate.net/publication/323384291_Bipolar_Junction_Transistor

E-content:

V. Milovanovic, R. van der Toorn, P. Humphries, D. P. Vidal and A. Vafanejad, "Compact model of Zener tunneling current in bipolar transistors featuring a smooth transition to zero forward bias current," *2009 IEEE Bipolar/BiCMOS Circuits and Technology Meeting*, 2009, pp. 99-102, doi: 10.1109/BIPOL.2009.5314134. https://ieeexplore.ieee.org/document/5314134

M. Oueslati, H. Garrab, A. Jedidi and K. Besbes, "The advantage of silicon carbide material in designing of power bipolar junction transistors," 2015 IEEE 12th International Multi-Conference on Systems, Signals & Devices (SSD15), 2015, pp. 1-6. https://ieeexplore.ieee.org/document/7348149

H. Luo, F. Iannuzzo, F. Blaabjerg, X. Wang, W. Li and X. He, "Elimination of bus voltage impact on temperature sensitive electrical parameter during turn-on transition for junction temperature estimation of high-power IGBT modules," *2017 IEEE Energy Conversion Congress and Exposition (ECCE)*, 2017, pp. 5892-5898 https://ieeexplore.ieee.org/document/8096974

F. Bauer, I. Nistor, A. Mihaila, M. Antoniou and F. Udrea, "Super junction IGBT Filling the Gap Between SJ MOSFET and Ultrafast IGBT," in *IEEE Electron Device Letters*, vol. 33, no. 9, pp. 1288-1290, Sept. 2012 https://ieeexplore.ieee.org/document/6246672

https://presiuniv.knimbus.com/user#/home

Topics relevant to “SKILL DEVELOPMENT”: Electrical & Electronic component and laws, Fundamentals of Digital Electronics, Communication Systems, Microprocessors and Computer Organization for Skill Development through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.

ourse Code: ENG1002	**ourse Title: Technical English** **ype of Course:1] School Core** **2] Laboratory integrated**	**L-T-P-C**	1-0-2-2
ersion No.	1.0 V. 3		
ourse Pre-requisites	ntermediate Level English		
ourse nti-requisites	IL		
ourse Description	Technical English course is designed to equip students with the language skills necessary for effective communication in technical and scientific contexts. The course focuses on the specialized vocabulary, writing styles, and communication techniques used in various technical fields, including engineering and information technology.		
ourse Objectives	The objective of this course is to develop the learners’ **EMPLOYABILITY SKILLS** by using **EXPERIENTIAL LEARNING** and **PARTICIPATIVE LEARNING TECHNIQUES.**		

Course Outcomes	**On successful completion of the course, the students shall be able to:** 1. Develop proficiency in using technical vocabulary and terminology. 2. Apply language skills for better speaking skills in technical fields. 3. Write technical descriptions 4. Demonstrate writing skills in writing technical documents such as reports, manuals, and articles.

Course Content:				
Module 1	**Fundamentals of Technical Communication**	**Worksheets& Quiz**	**Vocabulary building**	**Classes**
Introduction to Technical English Differences between Technical English and General English Technical Writing Basics Technical Vocabulary				
Module 2	**Technical Presentation**	**Presentations**	**Speaking Skills**	**2 Classes**
Introduction Planning the Presentation Creating the Presentation Giving the Presentation				
Module 3	**Technical Description**	**Assignment**	**Group Presentation**	**2 Classes**
Product Description Process Description User Manuals Transcoding: Diagrams, charts and images				
Module 4	**Technical Writing**	**Assignment**	**Writing Skills**	**2 Classes**
Email Writing Persuasive and Descriptive Language Professional Email Etiquette Writing clear and concise technical emails Communicating technical information effectively **Technical Report Writing** Types of technical reports (Lab reports, research reports, etc.) Components of technical reports Writing an abstract and executive summary Structure and content organization				

Transcoding: diagrams, charts and images

ist of Laboratory Tasks:

1. Module-1

evel 1: Worksheets

evel 2: Worksheets

2. Module 2

evel 1: Preparing Presentation

evel 2: Giving Presentation (Individual)

3. Module-3

evel 1: Product Description & User Manual

evel 2: Process Description & Transcoding

4. Module 4

evel 1: Email Writing

evel 2: Report Writing

argeted Applications & Tools that can be used:

1. Flipgrid
2. Quizzes
3. Youtube Videos
4. Podcast

roject work/Assignment: Mention the Type of Project /Assignment proposed for this course

Bring out the essence of technical communication with reference to the conventions of technical communication, with examples

Prepare a technical presentation on the importance of Technical Communication and its relevance in a technical field, with real-life examples.

he following individual, as well as group Assignments, will be given to the students.

1. **Presentation**
2. **Describing a product/process**
3. **Individual Reports**

ext Books

1. Kumar, Sanjay; Pushpalatha. *English Language and Communication Skills for Engineers.* Oxford University Press. 2018.
2. Brieger, Nick and Alison Paul. *Technical English Vocabulary and Grammar*. https://nmetau.edu.ua/file/technical_english_vocabulary_and_grammar.pdf

eference Book:

Chauhan, Gajendra Singh, and Kashmiramka, Smita, ***Technical Communication.*** Cengage Publication. 2018.

Sunder Jain. *Technical Report Writing*. Centrum Press, 2013.

John Bowden. "Writing a Report: How to Prepare, Write & Present Really Effective Reports?". 9th Edition 2011

omfort, Jeremy et. al. 1984. *Business Reports in English*. Cambridge University Press.

Sharma, R.C. and K. Mohan. 2011. Business Correspondence and Report Writing, Fourth Edition. Tata McGraw Hill.

Web Resources: ;https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=JSTOR1_3307. ;https://puniversity.informaticsglobal.com:2282/ehost/detail/detail?vid=5&sid=3a77d69b-abe5-4681-b39d-32dfdcb8f4a5%40redis&bdata=JnNpdGU9ZWhvc3QtbGl2ZQ%3d%3d#AN=154223466&db=iih : Last,Suzan, et. al. ***Technical Writing Essentials.*** University of Victoria, British Columbia, 2019 (E- Book) Wambui, Tabita Wangare, et al. ***Communication Skills- Volume 1,*** LAP LAMBRET, USA, 2012 (E Book)
opics Relevant to the Development of Employability Skills: **peaking Skills, Writing Skills, Critical Thinking and Critical Analysis, and Group Communication.**

urse Code: S 1001	**urse Title: Introduction to Soft Skills** **pe of Course: Practical Only Course**	T-P- C	0-0-2-1
rsion No.	1.0		
urse Pre-requisites	Students are expected to understand Basic English. Students should have desire and enthusiasm to involve, participate and learn.		
ti-requisites	NIL		
urse Description	This course is designed to enable students understand soft skills concepts and improve confidence, communication and professional skills to give the students a competitive advantage and increase chances of success in the professional world. The course will benefit learners in presenting themselves effectively through various activities and learning methodologies.		
urse Objective	The objective of the course is to **familiarize the learners with the concepts of "Soft Skills" and attain SKILL DEVELOPMENT through PARTICIPATIVE LEARNING techniques.**		
urse Out Comes	**On successful completion of this course the students shall be able to:** CO1: **Recognize** significance of soft skills CO2: **Illustrate** effective communication while introducing oneself and others CO3: **List** techniques of forming healthy habits CO4: **Apply** SMART technique to achieve goals and increase productivity		
urse Content:			

odule 1	TRODUCTION TO SOFT SKILLS		**ssroom activity**	**04 Hours**
Topics: Setting Expectations, Ice Breaker, Significance of soft skills, Formal grooming, punctuality				
odule 2	FECTIVE COMMUNICATION		**dividual Assessment**	**10 Hours**
Topics: Different styles of communication, Difference between hearing and listening, Effective communication for success, Email etiquette, Self-introduction framework, Video introduction, email- writing, Resume Building- Digital, Video, Traditional.				
odule 3	BIT FORMATION		**orksheets & Assignment**	**4 Hours**
Topics: Professional and personal ethics for success, Identity based habits, Domino effect, Habit Loop, Unlearning, standing up for what is right				
odule 4	al setting & Time Management		**al sheet**	**8 Hours**
A session where students will be introduced to Time management, setting SMART Goals, Introduction to OKR Techniques, Time Management Matrix, steps to managing time through outbound group activity, making a schedule, Daily Plan and calendars (To Do List), Monitoring/charting daily activity				
Targeted Application & Tools that can be used: LMS				
Project work/Assignment: Mention the Type of Project /Assignment proposed for this course				
1) Individual Assessment 2) LMS MCQ				
The topics related to Skill Development: Communication and professional grooming, Goal setting and presentation for skill development through participative learning techniques. This is attained through assessment component mentioned in course handout.				

Course Code: CSE1004	Course Title: Problem Solving Using C Type of Course: School Core Lab Integrated.	L- T-P-C	1	0	4	3
Version No.	1.0					

Course Pre-requisites	NIL			
Anti-requisites	NIL			
Course Description	The course is designed to provide complete knowledge of C language. Students will be able to develop logics which will help them to create programs and applications in C. AC Also by learning the basic programming constructs they can easily switch over to any other language in future.			
Course Object	The objective of the course is to familiarize the learners with the concepts of Problem Solving Using C and attain Employability through Problem Solving Methodologies.			
Course Outcomes	On successful completion of this course the students shall be able to: 1. Write algorithms and to draw flowcharts for solving problems 2. Demonstrate knowledge and develop simple applications in C programming constructs 3. Develop and implement applications using arrays and strings 4. Decompose a problem into functions and develop modular reusable code 5. Solve applications in C using structures and Union 6. Design applications using Sequential and Random Access File Processing.			
Course Content:				
Module 1	Introduction to C Language	Quiz	Problem Solving	9 Hrs.
Topics: Introduction to Programming – Algorithms – Pseudo Code - Flow Chart – Compilation – Execution – Preprocessor Directives (#define, #include, #undef) - Overview of C – Constants, Variables and Data types – Operators and Expressions – Managing Input and Output Operations – Decision Making and Branching - Decision Making and Looping.				
Module 2	Introduction to Arrays and Strings	Quiz	Problem Solving	9 Hrs.
Topics: Arrays: Introduction – One Dimensional Array – Initialization of One Dimensional Arrays – Example Programs – Sorting (Bubble Sort, Selection Sort) – Searching (Linear Search) - Two Dimensional Arrays – Initialization of Two Dimensional Arrays. Example Programs – Matrix operations. Strings: Introduction – Declaring and Initializing String Variables – Reading Strings from Terminal – Writing String to Screen – String Handling Functions.				
Module 3	Functions and Pointers	Quiz	Problem Solving	9 Hrs.
Topics: Functions: Introduction – Need for User-defined functions – Elements of User-Defined Functions: declaration, definition and function call–Categories of Functions – Recursion. Pointers: Introduction – Declaring Pointer Variables – Initialization of Variables – Pointer Operators – Pointer Arithmetic – Arrays and Pointers – Parameter Passing: Pass by Value, Pass by Reference.				
Module 4	Structures and Union	Quiz	Problem Solving	9 Hrs.
Topics: Structures: Introduction – Defining a Structure – Declaring Structure Variable – Accessing Structure Members – Array of Structures – Arrays within Structures – Union: Introduction – Defining and Declaring Union – Difference Between Union and Structure.				

Module 5	File handling	Case Study	Problem Solving	9 Hrs.

Topics:
Files: Defining and Opening a File – Closing a File – Input / Output Operations on File – Random Access Files

List of Practical
Tasks Lab Sheet
1 (Module I)
CHE1018
Lab Sheet 2 (Module II)
Programs using Arrays and Strings
Lab Sheet 3 (Module III)
Programs using Functions and Pointers
Lab Sheet 4 (Module IV)
Programs using Structures and Unions
Lab Sheet 5 (Module V)
Programs using Files

Text Book(s):

1. E. Balaguruswamy, “Programming in ANSI C”, 8th Edition, 2019, McGraw Hill Education, ISBN: 978-93-5316- 513-0.

Reference Book(s):

1. Yashwant Kanetkar, Let us C, 17th Edition, BPB Publications, 2020.
2. ReemaThareja, “Programming in C”, Oxford University Press, Second Edition, 2016.
3. Kernighan, B.W and Ritchie,D.M, “The C Programming language”, Second Edition, Pearson Education, 2015
4. Schildt Herbert, “C: The Complete Reference”, Tata McGraw Hill Education, 4th Edition, 2014.
5. Stephen G. Kochan, “Programming in C”, Addison-Wesley Professional, 4th Edition, 2014.

Web Links and Video Lectures:

1. https://nptel.ac.in/courses/106/105/106105171/
2. https://archive.nptel.ac.in/courses/106/104/106104128/

Course Code: CHE1018	**Course Title: Environmental Science** **Type of Course: School Core- Theory and Lab**	**L- T-P- C**	1	0	2	0
Version No.	2.0					
Course Pre-requisites	NIL					
Anti-requisites	**NIL**					

Course Description	This course emphasizes the need to conserve biodiversity and adopt a more sustainable lifestyle by utilizing resources in a responsible way. Topics covered include basic principles of ecosystem functions; biodiversity and its conservation; human population growth; water resources, pollution; climate change; energy resources, and sustainability; Sustaining human societies, policies, and education. **This course is designed to cater to Environment and Sustainability**			
Course Objective	The objective of the course is to **familiarize the learners with the concepts of "Environmental Science" and attain SKILL DEVELOPMENT through EXPERIENTIAL LEARNING techniques.**			
Course Outcomes	On successful completion of this course the students shall be able to: Appreciate the historical context of human interactions with the environment and the need for eco-balance. Describe basic knowledge about global climate change with particular reference to the Indian context. Understand biodiversity and its conservation Develop an understanding on types of pollution and ways to protect the environment Learn about various strategies on Global environmental management systems			
Course Content:				
Module 1	**Humans and the Environment**	Assignment	Data Collection	**01 class**
Topics: The man-environment interaction: Mastery of fire; Origin of agriculture; Emergence of city states; Great ancient civilizations and the environment. ***Self-learning topics:*** Humans as hunter-gatherers; Industrial revolution and its impact on the environment; Environmental Ethics and emergence of environmentalism.				
Module 2	**Natural Resources and Sustainable Development**	Assignment		**03 Classes**
Topics: Overview of natural resources: Definition of resource; Classification of natural resources- biotic and abiotic, renewable and non-renewable. **Water resources**: Types of water resources- fresh water and marine resources; **Soil and mineral resources**: Important minerals; Mineral exploitation Soil as a resource and its degradation. **Energy resources**: Sources of energy and their classification, renewable and non-renewable sources of energy; Advantages and disadvantages. ***Self- learning topics:*** Availability and use of water resources; Environmental impact of over-exploitation, issues and challenges.; Environmental problems due to extraction of minerals and use; Sustainable Development Goals (SDGs)- targets, indicators, and challenges for SDGs.				
Module 3	**Environmental Issues: Local, Regional and Global**	Case study		**02 Classes**
Topics: **Environmental Pollution**: Types of Pollution- air, noise, water, soil, municipal solid waste, hazardous waste; Trans- boundary air pollution; Acid rain; Smog. **Land use and Land cover change**: land degradation, deforestation, desertification, urbanization. Global change: Ozone layer depletion; Climate change ***Self -learning topics:*** Environmental issues and scales				

	Module 4	**Conservation of Biodiversity and Ecosystems**	Assignment		**02 Classes**

Topics:

Biodiversity-Introduction, types, Species interactions, Extinct, endemic, endangered and rare species, Threats to biodiversity: Natural and anthropogenic activities.

Self-learning topics: Mega-biodiversity, Hot-spots, Major conservation policies. Biodiversity loss: past and current trends, impact.

	Module 5	**Environmental Pollution and Health**	Case study		**03 Classes**

Topics:

Pollution, Definition, point and nonpoint sources of pollution, **Air pollution**- sources, major air pollutants, health impacts of air pollution.

Water pollution– Pollution sources, adverse health impacts on human and aquatic life and mitigation, Water quality parameters and standards.

Soil pollution and solid waste- Soil pollutants and their sources, solid and hazardous waste, Impact on human health.

Self-learning topics: Noise pollution, Thermal and radioactive pollution.

	Module 6	**Climate Change: Impacts, Adaptation and Mitigation**	Assignment/case		**02 Classes**

Topics:

Understanding climate change: Natural variations in climate; Projections of global climate change with special reference to temperature, rainfall and extreme events; Importance of 1.5 °C and 2.0 °C limits to global warming; Impacts

Vulnerability and adaptation to climate change: Observed impacts of climate change on ocean and land systems; Sea level rise, changes in marine and coastal ecosystems; Impacts on forests and natural ecosystems; Indigenous knowledge for adaptation to climate change.

Self-learning topics: Mitigation of climate change: Synergies between adaptation and mitigation measures; National and international policy instruments for mitigation.

	Module 7	**Environmental Management**	Case study	Data analysis	**02 Classes**

Topics:

Environmental management system: ISO 14001; Environmental risk assessment Pollution control and management; Waste Management- Concept of 3R (Reduce, Recycle and Reuse) and sustainability.

Self-learning topics: Environmental audit and impact assessment; Eco labeling /Eco mark scheme

	Module 8	**Environmental Treaties and Legislation**	Case study	Data analysis	**01 Classes**

Topics:

Major International Environmental Agreements: Convention on Biological Diversity (CBD), Major Indian Environmental Legislations: Environmental Protection Act, Forest Conservation Act, Public awareness.

Self-learning topics: Paris Agreement, Conference of the Parties (COP), India's status as a party to major conventions: Air (Prevention and Control of Pollution) Act, Water (Prevention and control of Pollution) Act, Wildlife Protection Act.

List of laboratory tasks : Any eight experiments will be conducted

ermination of total alkalinity of a water sample (knowledge)

Estimation of water hardness by EDTA method and its removal (by zeolite/ ion exchange method) (Comprehensive)

mation of copper from industrial effluents by colorimetric method (Comprehensive)

mation of iron from industrial effluents by titrimetric method/potentiometric method (Comprehensive)

mation of nickel from industrial effluents by titrimetric method (Comprehensive)

mation of chloride in drinking water by titrimetric method (Comprehensive)

mation of fluoride in ground water by colorimetric method (Comprehensive)

ermination of calcium in aqueous solution (Comprehensive)

ermination of Total Dissolved Salts, conductivity and pH of a water samples (Knowledge)

Determination of Chemical oxygen demand in the industrial effluent. (Comprehensive)

Biological oxygen demand of waste water sample (Comprehensive)

Determination of dissolved oxygen of an industrial effluent (Comprehensive)

Quality monitoring analysis of a soil sample (knowledge)

Flame photometric estimation of Sodium and potassium (Application)

Gas Chromatographic analysis of volatile organic compounds (Application)

Targeted Application & Tools that can be used:

Application areas are Energy, Environment and sustainability

Tools: Statistical analysis of environmental pollutants using excel, origin etc.

Project work/Assignment:

Assessment Type

term exam

Assignment (review of digital/ e-resource from PU link given in references section - mandatory to submit screenshot accessing the digital resource.)

evaluation/Assignment

Term Exam

-learning

Assignment 1: Write a Statement of Environment report of your town/city/state/country

Assignment 2: Individual students will carry out the analyses of polluted solid, liquid, and gaseous samples and propose suitable mitigation measures. A detailed and in-depth report needs to be submitted for each case. This may include preparation of reagents, sample preparation (extraction), chemical analysis carried out, instruments and tools used, data collected and processed, inferences made and conclusions arrived at. Necessary support is given in the form of
lab manual and reference links to e-books.

Text Book

yler Miller and Scott Spoolman (2020), Living in the Environment, 20th Edition, Cengage Learning, USA

nnamurthy, K.V. (2003) Text book of Biodiversity, Science Publishers, Plymouth, UK.

Jackson, A.R. & Jackson, J.M. (2000), Environmental Science: The natural environment and human impact, Pearson Education.

Reference Books

Fisher, Michael H. (2018) An Environmental History of India- From Earliest Times to the Twenty-First Century, Cambridge University Press.

William P. Cunningham and Mary Ann Cunningham (2017), Principles of Environmental Science: Inquiry & Applications, 8th Edition, McGraw-Hill Education, USA.

a N., (2020) Wild and Wilful. Harper Collins, India.

www.ipcc.org; https://www.ipcc.ch/report/sixth-assessment-report-cycle/

Theodore, M. K. and Theodore, Louis (2021) Introduction to Environmental Management, 2nd Edition. CRC Press.

Richard A. Marcantonio, Marc Lame (2022). Environmental Management: Concepts and Practical Skills. Cambridge University Press.

E-resources:

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DOAB_1_06082022_18126
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DOAB_1_06082022_8761
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DOAJ_1_02082022_3333
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DOAB_1_06082022_3063
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DOAB_1_06082022_20719
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DOAB_1_06082022_16824
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DOAB_1_06082022_3954
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DOAB_1_06082022_491
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=CUSTOM_PACKAGE_16012023_WORLD_BUSINESS_COUNCIL_SUSTAINABLE_488
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=CUSTOM_PACKAGE_16012023_WORLD_BUSINESS_COUNCIL_SUSTAINABLE_583
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=SPRINGER_INDEST_1_171
https://presiuniv.knimbus.com/user#/searchresult?searchId=3R%20principle&_t=1687427221129
https://presiuniv.knimbus.com/user#/searchresult?searchId=eco%20labelling&_t=1687427279979
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=TEXTBOOK_LIBRARY01_06082022_395&xIndex=4
https://www.ugc.gov.in/oldpdf/modelcurriculum/env.pdf

Topics relevant to Skill Development:

Industrial revolution and its impact on the environment, Environmental impact of over-exploitation of water resources, pollution and ill effects, lab experiments for Skills development through Problem solving Techniques. This is attained through assessment component mentioned in course handout.

All topics in theory component are relevant to Environment and Sustainability.

<table>
<tr><td>Course Code:
PPS 1011</td><td colspan="3">Course Title: Introduction to Verbal

Ability Type of Course: Theory Only

Course</td><td>L- T- P- C</td><td>0</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>Version No.</td><td colspan="8">1.0</td></tr>
<tr><td>Course Pre-requisites</td><td colspan="8">Students are expected to understand Basic English.

Students should have desire and enthusiasm to involve, participate and learn.</td></tr>
<tr><td>Anti-requisites</td><td colspan="8">NIL</td></tr>
<tr><td>Course Description</td><td colspan="8">This course is designed to enable students understand the importance of Verbal Ability and improve confidence, communication and professional skills to give them a competitive advantage and increase chances of success in the professional world. The course will benefit learners in presenting themselves effectively through various worksheets and learning methodologies.</td></tr>
<tr><td>Course Objective</td><td colspan="8">The objective of the course is to familiarize the learners with the concepts of “Verbal Ability” and attain SKILL DEVELOPMENT through PARTICIPATIVE LEARNING techniques.</td></tr>
<tr><td>Course Out Comes</td><td colspan="8">On successful completion of this course the students shall be able to:

CO1: Recognize significance of verbal

ability CO2: Utilize the rules of

communication

CO3: Apply techniques of vocabulary building to showcase effective communication</td></tr>
<tr><td>Course Content:</td><td colspan="8"></td></tr>
<tr><td>Module 1</td><td colspan="3">INTRODUCTION TO VERBAL ABILITY</td><td colspan="3">Individual Assessment</td><td colspan="2">01 Hour</td></tr>
<tr><td colspan="9">Topics: Setting Expectations, Ice Breaker, Significance of verbal ability, pre-assessment</td></tr>
<tr><td>Module 2</td><td colspan="3">EFFECTIVE VERBAL COMMUNICATION</td><td colspan="3">Practice Worksheets</td><td colspan="2">06 Hours</td></tr>
<tr><td colspan="9">Topics: Different rules of grammar and application, Subject-Verb Agreement, Tenses</td></tr>
<tr><td>Module 3</td><td colspan="3">VOCABULARY BUILDING</td><td colspan="3">Practice Worksheets</td><td colspan="2">04 Hours</td></tr>
</table>

Topics: Root words, Synonyms and antonyms, analogies, para-jumbles			
Module 4	READING COMPREHENSION	**Individual Assessment**	**02 Hours**
A session where students will be introduced to speed reading and comprehension, **post-assessment**			
Targeted Application & Tools that can be used: LMS			
Project work/Assignment: Mention the Type of Project /Assignment proposed for this course			
Individual Assessment LMS MCQ			
The topics related to Skill Development: Communication, grammar rules, vocabulary building, effective presentation for skill development through participative learning techniques. This is attained through **learning and practicing the rules of effective communication through worksheets** as mentioned in the assessment component.			

urse Code: AT1003	**urse Title: Applied Statistics** **pe of Course:** School Core	**. T P C**	1	0	2	2
rsion No.	)					
urse Pre-requisites	ne					
ti-requisites	ne					
urse Description	The goal of this course is to provide a firm understanding of probability and statistics by means of a thorough treatment of descriptive statistics, probability and probability distributions keeping in mind the future courses having statistical, quantitative and probabilistic components. The course covers topics such as descriptive statistics, probability, rules for probability, random variables and probability distributions, standard discrete and continuous probability distributions.					
urse Objective	The objective of the course is to **familiarize the learners with the concepts of "**Applied Statistics" and attain **Skill Development Through Problem Solving techniques**.					
pected Outcome:	At the end of this course, students will be in a position to apply the techniques of descriptive statistics effectively interpret the ideas of probability and conditional probability demonstrate the knowledge of probability distributions Compute statistical parameters, correlation and regression, probability and sampling distributions using R software.					
dule 1	**scriptive Statistics**	signment	ding needed			**10 classes**
Introduction to Statistics, Data and statistical thinking, review of basic statistical parameters, Covariance, Correlation, Types of Measures of Correlation - Karl Pearson's Correlation Coefficient, Spearman Rank Correlation, linear regression, Multi linear regression **.**						
dule 2	**bability**					**6 classes**
roduction to Probability, Probability of an event, Addition Principle, Multiplication law, Conditional Probability, Total Probability and Baye's theorem with examples						
dule 3	**ndom Variables and Probability Distributions**		ding needed			**14 classes**
Introduction to Random variables, Discrete Random Variables and Continuous Random Variables, Probability Distributions, Probability Mass Function and Probability Density Function, Various Probability distributions, Binomial, **Negative Binominal (Self Study)**, Poisson, Normal and Exponential distributions						
dule 4	**mpling Theory**		ding			**15 classes**

			needed	

Introduction to Sampling Theory, Population, Statistic, Parameter, Sampling Distribution, Standard Error. Testing of Hypothesis, Types of Errors, Critical Region, level of Significance. Difference between Parametric and Non-parametric Tests, Large Sample Tests: Z-Test for Single Mean and **Difference of Means (Self Study)**, Small Sample Tests: Student's t-Test for Single Mean and **Difference of Means,** F-Test, Chi-Square Test.

rgeted Application & Tools that can be used:

The objective of the course is to familiarize students with the theoretical concepts of probability and statistics and to equip them with basic statistical tools to tackle engineering and real-life problems.

ols used: R Software / MS-Excel

xt Book

1. Ronald E Walpole, Raymond H Myers, Sharon L Myers, and Keying E Ye, Probability and Statistics for Engineers and Scientists, Pearson Education, 2016.

ferences

1. James T. McClave, P. George Benson and Terry Sincich, Statistics for Business and Economics, 2018.
2. David R. Anderson, Dennis J. Sweeney, Thomas A. Williams, Essentials of Modern Business Statistics with Microsoft Excel, 2020.
3. David R. Anderson, Dennis J. Sweeney, Thomas A. Williams, Essentials of Statistics for Business and Economics, 2019.
4. Douglas C. Montgomery and George C. Runger, Applied Statistics and Probability for Engineers, John Wiley and Sons, 2018.
5. Richard A. Johnson, Miller and Freund's Probability and Statistics for Engineers, 2018.
6. Kishor S Trivedi, Probability and Statistics with reliability, Queuing and Computer Science Applications, John Wiley & Sons, 2008.

pics relevant to SKILL DEVELOPMENT: The goal of this course is to provide a firm understanding of probability and statistics by means of a thorough treatment of descriptive statistics, probability and probability distributions keeping in mind the future courses having statistical, quantitative and probabilistic components. The course covers topics such as descriptive statistics, probability, rules for probability, random variables and probability distributions, standard discrete and continuous probability distributions for **Skill Development through Problem Solving methodologies.** This is attained through assessment component mentioned in course handout.

urse Code: E2007	**urse Title: Digital Design pe of Course: Theory &Integrated Laboratory**	**L- T-P- C**	**2**	**0**	**2**	**3**
rsion No.						

urse Pre-requisites	**[1] Elements of Electronics/Electrical Engineering, 2] Basic concepts of number representation, Boolean Algebra**
ti-requisites	
urse Description	**The purpose of this course is to enable the students to appreciate the fundamentals of digital logic circuits and Boolean algebra focusing on both combinational and sequential logic circuits. The course emphasizes on minimization techniques for making canonical and low-cost digital circuit implementations. This course deals with analysis and design of digital electronic circuits. The course also creates a foundation for future courses which includes Computer Architecture, Microprocessors, Microcontrollers, and Embedded Systems etc.** The course enhances the Design, Implementation and Programming abilities through laboratory tasks. The associated laboratory provides an opportunity to verify the theoretical knowledge.
urse Objective	**The objective of the course is to familiarize the learners with the concepts of Digital Design and attain the SKILL DEVELOPMENT through** EXPERIENTIAL **LEARNING.**
urse Outcomes	**On successful completion of this course the students shall be able to:** i. **Describe** the concepts of number systems, Boolean algebra and logic gates. ii. **Apply** minimization techniques to simplify Boolean expressions. iii. **Demonstrate** the Combinational circuits for a given logic iv. Demonstrate the Sequential and programmable logic circuits v. **Implement** various combinational and sequential logic circuits using gates.
urse Content:	

dule 1	**ndamentals of Number systems- Boolean algebra and digital logic**	plication Assignment	ta Analysis task	**06 classes**
pics: Review of Number systems and logic gates, Number base conversions, Overview of Boolean functions and simplifications, two, three, four variable K-Maps- Don't care conditions- Both SOP and POS- Universal Gates (NAND & NOR) Implementations. Introduction to HDL.				
dule 2	**Boolean function simplification**	plication Assignment	ta Analysis task	**08 Classes**
pics: Introduction to Combinational circuits, Analysis, Design procedure, Binary Adder and Subtractor, Magnitude comparator, Parity generator and checker, Multiplexers-Demultiplexers, Decoders, Encoders and Priority Encoders, HDL Models of combinational circuits.				
dule 3	**Combinational Logic circuits:**	plication Assignment	gramming Task & Data Analysis task	**08 Classes**
pics: Introduction to sequential circuits, Storage elements: latches and flip flops, Characteristic tables and equations, excitation table, Analysis of clocked sequential circuits, Mealy & Moore Models of finite state machines - Registers & Counters. HDL Models of Sequential circuits.				

t of Laboratory Tasks:

periment N0 1: Verify the Logic Gates truth table

vel 1: By using Digital Logic Trainer kit

vel 2: By using Analog devices like RPS, Volt meter, Resistors and ICs

periment No. 2: Verify the Boolean Function and Rules
vel 1: By using Digital Logic Trainer kit
vel 2: By using Analog devices like RPS, Volt meter, Resistors and ICs

periment No. 3: Design and Implementations of HA/FA
vel 1: By using basic logic gates and Trainer Kit
vel 2: By using Universal logic gates and Trainer Kit

periment No. 4: Design and Implementations of HS/FS
vel 1: By using basic logic gates and Trainer Kit
vel 2: By using Universal logic gates and Trainer Kit

periment No. 5: Design and Implementations of combinational logic circuit for specifications
vel 1: Specifications given in the form of Truth table
vel 2: Specification should be extracted from the given scenario

periment No. 6: Study of Flip flops

periment No. 7: Design and Implementations of sequential logic circuit for specifications
vel 1: Specifications given in the form of Truth table
vel 2: Specification should be extracted from the given scenario

periment No.8: HDL coding for basic combinational logic circuits
vel 1: Gate level Modeling
vel 2: Behavioral Modeling

periment No.9: HDL coding for basic sequential logic circuit
vel 1: Gate level Modeling
vel 2: Behavioral Modeling

rgeted Application & Tools that can be used:

gital electronics is the foundation of all modern electronic devices such as cellular phones, MP3 players, laptop computers, digital cameras, high definition televisions, Home Automation, Communication in systems in industries

ofessionally Used Software: HDL/VHDL/Verilog HDL/ OOPS

Text Book(s):

1. Mano, M. Morris and Ciletti Michael D., *"Digital Design"*, Pearson Education, 6th edition
2. Thomas L. Floyd "DIGITAL LOGIC DESIGN" , Pearson Education, fourth edition.

Reference(s):

Reference Book(s):

R1. Jain, R. P., "Modern Digital Electronics", McGraw Hill Education (India), 4th Edition

R2. Roth, Charles H., Jr and Kinney Larry L., "Fundamentals of logic Design", Cengage Learning, 7th Edition

Online Resources (e-books, notes, ppts, video lectures etc.): Book Free Download (studymaterialz.in)

1. **eBook1**: Mano, M. Morris and Ciletti Michael D., *"Digital Design"*, Pearson Education.
2. {[PDF] Digital Design By M. Morris Mano, Michael D Ciletti Book Free Download }
3. **eBook2:**Floyd "DIGITAL LOGIC DESIGN" fourth edition- ePub, eBook- [PDF] DIGITAL LOGIC DESIGN FOURTH EDITION FLOYD | abri.engenderhealth.org.
4. NPTEL Course- NPTEL :: Electrical Engineering - NOC:Digital Electronic Circuits
5. Digital Logic Design PPT Slide 1 (iare.ac.in)
6. Lab Tutorial: Multisim Tutorial for Digital Circuits - Bing video

CircuitVerse - Digital Circuit Simulator online

Learn Logisim➠ Beginners Tutorial | Easy Explanation! - Bing video

Digital Design 5: LOGISIM Tutorial & Demo

7. https://presiuniv.knimbus.com/user#/home

E-content:

1. Z. Xin-Li and W. Hong-Ying, "The Application of Digital Electronics in Networking Communication," 2016 Eighth International Conference on Measuring Technology and Mechatronics Automation (ICMTMA), 2016, pp. 684-687, doi: 10.1109/ICMTMA.2016.168.
2. An encoding technique for design and optimization of combinational logic circuit DipayanBhadra;Tanvir Ahmed Tarique;Sultan Uddin Ahmed;Md. Shahjahan;KazuyukiMurase2010 13th International Conference on Computer and Information Technology (ICCIT)
3. A. Matrosova and V. Provkin, "Applying Incompletely Specified Boolean Functions for Patch Circuit Generation," 2021 IEEE East-West Design & Test Symposium (EWDTS), 2021, pp. 1-4, doi: 10.1109/EWDTS52692.2021.9581029.
4. A. Matrosova, V. Provkin and E. Nikolaeva, "Masking Internal Node Faults and Trojan Circuits in Logical Circuits," *2019 IEEE East-West Design & Test Symposium (EWDTS)*, 2019, pp. 1-4, doi: 10.1109/EWDTS.2019.8884434.

Topics relevant to "SKILL DEVELOPMENT": Adders, Multiplexers, Decoders / Encoders; Flip-Flops, Counters and Registers for **Skill Development** through **Experiential Learning** techniques. This is attained through assessment component mentioned in course handout.

urse Code: /1008	**urse Title: Basic Engineering Sciences pe of Course: Theory Only**	**-P-C**	2	0	0	2
rsion No.	)					
urse Pre-requisites	**NIL**					
ti-requisites	L					

urse Description	This basic course on engineering science is designed to introduce students to the fields of civil, mechanical and petroleum engineering. Student will be exposed to various fields in civil engineering and different manufacturing techniques in addition to machinery for power production and consumption. Additionally, students will be getting an overview of various sectors of oil & gas industries. This course acquaints students to basics of Industry 4.0 and Construction 4.0. The course aims to enable students to appreciate the multidisciplinary nature of engineering design and operations in the current era with mechanization and digitization transforming every aspect of engineering.			
urse Objective	The objective of the course is skill development of student by using Participative Learning techniques.			
urse Outcomes	On successful completion of this course the students shall be able to: 1] Recognize the significance of various disciplines in Civil Engineering 2] Discuss the recent evolutions in Civil Engineering 3] Explain various energies, energy generating machineries and energy consumption machineries 4] Describe the fundamental concept and terminology associated with the Petroleum Industry 5] Distinguish between conventional and modern manufacturing techniques.			
urse Content:				
odule 1	**roduction to various fields in Civil Engineering**	ignment	se studies on different Civil Engineering Projects	**essions**
pics: Introduction to Civil Engineering: Definition, scope and branches of Civil Engineering, Role of Civil Engineer, Overview of Infrastructure.				
odule 2	**rrent Trends and Evolution in Civil Engineering**	ignment	ticle Review	**essions**
pics: Mechanization in Construction, Application of Digital Technologies in Planning, Design, execution, monitoring and maintenance of Construction. Overview of Smart Cities.				
odule 3	**wer Production and Consumption Machinery**	ignment & Quiz	ta Collection	**essions**
pics: Energy and its types, Engines and their applications, Pumps-Compressors and their applications.				

odule 4	**erview of Petroleum Engineering**	ignment & Quiz	ticle Review	**essions**

erview of the Petroleum Industry, Importance of Petroleum Engineering, lifecycle of Petroleum products, Classifications of E&P activities: Key difference between Offshore and Onshore, Onshore facilities, offshore platforms, Digitization of petroleum engineering

odule 5	**dustry 4.0**	ignment & Quiz	ta Collection	**essions**

pics: Conventional manufacturing process: Metal forming, metal removal and metal joining process.

odern Manufacturing process: 3D Printing / Additive Manufacturing.

rgeted Application & Tools that can be used:

plication Areas include design and implementation of Smart City projects, Infrastructure maintenance, Power production, IC engines, Electric vehicles, onshore and offshore exploration and production activities

oject work/Assignment:

signment 1: Collect data and prepare report on various Mega Projects in Civil Engineering

signment 2: Review Articles on current evolutions in Civil Engineering.

signment 3: Collect data related to renewable energy generation (Wind, Solar)

signment 4: Prepare an energy consumption chart for a compressor or pumps.

signment 5: Prepare a report on role of 3D printing across various industries.

signment 6: Prepare an assignment on geopolitical influence on oil and gas industries.

xt Book:

T1. Elements of Civil and Mechanical Engineering, L.S. Jayagopal & R Rudramoorthy, Vikas Publishers

T2. Elements of Mechanical Engineering, by VK Manglik

T3. Fundamentals of Oil & Gas Industry for Beginners by Samir Dalvi, Notion Press; 1st edition

ferences

1. K.P. Roy, S.K. Hajra Choudhury, Nirjhar Roy, "Elements of Mechanical Engineering", Media Promoters and Publishers Pvt Ltd, Mumbai.
2. Nontechnical Guide to Petroleum Geology, Exploration, Drilling & Production by Norman J. Hyne, PennWell Books; 3rd Revised edition

eb-resources:

1. Basic Civil Engineering

https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=2706932&site=ehost-live

2. Post-parametric Automation in Design and Construction

https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=1155197&site=ehost-live

3. Smart Cities : Introducing Digital Innovation to Cities

https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=1993146&site=ehost-live

4. Innovation Energy: Trends and Perspectives or Challenges of Energy Innovation

https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=2323766&site=ehost-live

5. Mechanical Engineering
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO106_REDO_1705
6. Additive Manufacturing: Opportunities, Challenges, Implications
https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=1134464&site=ehost-live
7. Society of Petroleum Engineers (SPE)
https://www.spe.org/en/
8. PetroWiki: A comprehensive online resource created by the Society of Petroleum Engineers that provides information on various aspects of petroleum engineering.
https://petrowiki.spe.org/PetroWiki
9. Rigzone: A resource for news and information about the oil and gas industry, including job postings and industry trends.
https://www.rigzone.com/

pics relevant to the development of SKILLS:

Engines-Turbines and their applications.
echanization in Construction.
itization in Petroleum Industries

Course Code: MEC1006	**Course Title: Engineering Graphics** **Type of Course: School Core & Theory Only**	**L- T-P- C**	2-0-0-2
Version No.	1.2		
Course Pre-requisites	**NIL**		
Anti-requisites	**NIL**		
Course Description	The course is designed with the objective of giving an overview of engineering graphics. It is introductory in nature and acquaints the students with the techniques used to create engineering drawings. The course emphasizes on projection of points, lines, planes and solids and isometric projections.		

<table>
<tr><td>Course Objective</td><td colspan="4">The objective of the course is to familiarize the learners with the concepts of “Engineering Graphics” and attain SKILL DEVELOPMENT through Problem solving methodologies.</td></tr>
<tr><td>Course Outcomes</td><td colspan="4">On successful completion of this course the students shall be able to:

(1) Demonstrate competency of Engineering Graphics as per BIS conventionsand standards.
(2) Comprehend the theory of projection for drawing projections of Points, Lines andPlanes under different conditions.
(3) Prepare multiview orthographic projections of Solids by visualizing them indifferent positions.
(4) Prepare pictorial drawings using the principles of isometric projections to visualizeobjects in three dimensions.</td></tr>
<tr><td colspan="5">Course Content:</td></tr>
<tr><td>Module 1</td><td>Introduction to Drawing</td><td>Assignment</td><td>Standard technical drawing</td><td>02 Sessions</td></tr>
<tr><td colspan="5">Topics:
Introduction, drawing instruments and their uses, relevant BIS conventions and standards, Lettering, Lineconventions, dimensioning, Selection of drawing sheet size and scale.
[02 Hours: Comprehension Level]</td></tr>
<tr><td>Module 2</td><td>Orthographic projections of Points, Straight Linesand Plane Surfaces</td><td>Assignment</td><td>Projection methods Analysis</td><td>10 Sessions</td></tr>
<tr><td colspan="5">Topics:
Introduction, Definitions – Elements of projection and methods of projection, Planes of projection, reference line and conventions adopted. First angle and third angle projections. Projection of Points inall 4 quadrants.
Projections of Straight Lines (located in first quadrant/first angle projection only): True and apparent lengths, true and apparent Inclinations to reference planes. (No application problems). Projection of Plane surfaces (First angle projection): Regular plane surfaces – triangle, square, rectangle, pentagon, hexagon and circle – in different positions inclined to both the planes using change of position method only.
[10 Hours: Application Level]</td></tr>
<tr><td>Module 3</td><td>Orthographic Projections of Solids</td><td>Assignment</td><td>Multi-view drawing Analysis</td><td>10 Sessions</td></tr>
</table>

Topics: Introduction, Projection of right regular prisms, pyramids, cone, hexahedron and tetrahedron in different positions (Problems resting on HP only and First angle projection). **[10 Hours: Application Level]**				
Module 4	Isometric Projections of Solids (Using isometric scale only)	Assignment	Spatial Visualization	**8 Sessions**
Topics: Introduction, Isometric scale, Isometric projections of right regular prisms, cylinders, pyramids, cones and their frustums, spheres and hemispheres, hexahedron (cube), and combination of 2 solids, conversion of orthographic view to isometric projection of simple objects. **[8 Hours: Application Level]**				
Text Book: 1.N. D. Bhatt, "Engineering Drawing: Plane and Solid Geometry," Charotar Publishing House Pvt. Ltd.				
References: **1.** K.R. Gopalakrishna, "Engineering Graphics", Subhash Publishers, Bangalore. **2.** D. M. Kulkarni, A. P. Rastogi, A. K. Sarkar, "Engineering Graphics with AutoCAD," Prentice Hall. **3.** D. A. Jolhe, "Engineering Drawing with Introduction to AutoCAD," Tata McGraw Hill. **Web resources:** **https://nptel.ac.in/courses/112103019**				
Topics relevant to "SKILL DEVELOPMENT": Projection in first and third angle for **SKILL DEVELOPMENT** through **Problem Solving methodologies**. This is attained through the assessment component mentioned in the course handout.				

Course Code: CSE1006	**Course Title:** Problem Solving using JAVA **Type of Course: Lab Integrated**	**L- T-P-C**	1	0	4	3
Version No.	2.0					
Course Pre-requisites	CSE1004 – Problem-Solving Using C					
Anti-requisites	Nil					
Course Description	This course introduces the core concepts of object-oriented programming. This course has theory and lab component which emphasizes understanding the implementation and application of object-oriented programming paradigm. It helps the student to build real-time secure applications by applying these concepts and also for effective problem-solving. The students interpret and understand the need for object-oriented programming to build applications.					

Course Objective	The objective of the course is to familiarize the learners with the concepts of Problem-Solving using JAVA and attain **SKILL DEVELOPMENT** through **EXPERIENTIAL LEARNING** techniques			
Course Out Comes	**On successful completion of the course, the students shall be able to:** **C.O. 1:** Describe the basic programming concepts. [Knowledge] **C.O. 2:** Apply the concept of classes, objects and methods to solve problems. [Application] **C.O. 3:** Apply the concept of arrays and strings. [Application] **C.O. 4:** Implement inheritance and polymorphism in building secure applications. [Application] **C.O. 5:** Apply the concepts of interface and error handling mechanism. [Application]			
Course Content:				
Module 1	**Basic Concepts of Programming and Java**	Assignment	Data Collection/Interpretation	**12 Sessions**
Topics: Introduction to Principles of Programming: Process of Problem Solving, Java program structure, Download Eclipse IDE to run Java programs, Sample program, Data types, Identifiers, Variables, Constants in java, Operators, Assignments and Expression, Basic Input/ Output functions, Control Statements: Branching and Looping.				
Module 2	**Classes, objects, methods and Constructors**	Case studies / Case let	Case studies / Case let	**12 Sessions**
Topics: Classes, Objects and Methods: Introduction to object Oriented Principles, defining a class, adding data members and methods to the class, access specifiers, instantiating objects, reference variable, accessing class members and methods. Static Polymorphism: Method overloading, constructors, constructor overloading, this keyword, static keyword, Nested classes, Accessing members in nested classes.				
Module 3	**Arrays, String and String buffer**	Quiz	Case studies / Case let	**14 Sessions**
Topics: Arrays: Defining an Array, Initializing & Accessing Array, Multi–Dimensional Array, Array of objects. String: Creation & Operation. String builder class, methods in String Buffer.				
Module 4	**Inheritance and Polymorphism**	Quiz	Case studies / Case let	**14 Sessions**
Topics: Inheritance: Defining a subclass, Types of Inheritance, super keyword. Dynamic Polymorphism: Method overriding. Final keyword: with data members, with member functions and with class. Abstract keyword: with data members, with member functions and with class, Exception handling.				
Module 5	**Input & Output Operation in Java**	Quiz	Case studies / Case let	**14 Sessions**
Input/output Operation in Java(java.io Package), Streams and the new I/O Capabilities, Understanding Streams, working with File Objects, File I/O Basics, Reading and Writing to Files, Buffer and Buffer Management, Read/Write Operations with File Channel, Serializing Objects, Observer and Observable Interfaces.				
List of Laboratory Tasks:				

P1 - Problem Solving using Basic Concepts.
P2 - Problem Solving using Basic Concepts and Command Line Arguments.
P3 - Programming assignment with class, objects, methods and Constructors.
P4 - Programming assignment with method overloading.
P5 - Programming assignment with constructor overloading.
P6 - Programming assignment with Static members and static methods.
P7 - Programming assignment with Nested classes.
P8 - Programming assignment using Arrays.
P9 - Programming assignment using Strings.
P10 - Programming assignment using String Builder.
P11 - Programming assignment using Inheritance and super keyword.
P12 - Programming assignment using Method overriding and Dynamic method invocation.
P13 - Programming assignment using Final keywords.
P14 - Programming assignment using Abstract keywords.
P15 - Programming assignment using Interface.
P16 - Programming assignment using Interface.
P17 - Programming assignment CharacterStream Classes
P18 - Programming assignment Read/Write Operations with File Channel

Targeted Application & Tools that can be used : JDK /eclipse IDE/ net Beans IDE.

Text Book

T1 Herbert Schildt, "The Complete Reference Java 2", Tata McGraw Hill Education.

References

R1: Cay S Horstmann and Cary Gornell, "CORE JAVA volume I-Fundamentals", Pearson

R2: James W. Cooper, "Java TM Design Patterns – A Tutorial", Addison-Wesley Publishers.

E book link R1: http://rmi.yaht.net/bookz/core.java/9780134177373-Vol-1.pdf

E book link R2: Java(tm) Design Patterns: A Tutorial([PDF] [7qmsenjl97t0] (vdoc.pub)

Web **resources**

ps://youtube.com/playlist?list=PLu0W_9lII9agS67Uits0UnJyrYiXhDS6q

ps://puniversity.informaticsglobal.com:2229/login.aspx

Topics relevant to the development of "Skill Development":

1. Static Polymorphism
2. Method overloading, constructors
3. constructor overloading
4. this keyword
5. static keyword and Inner classes
6. Inheritance and Polymorphism.

for **Skill Development** through **Experiential Learning** techniques. This is attained through the assessment component mentioned in the course handout.

G2001	**vanced English**	**T- P- C**				2
rsion No.						
urse Pre-requisites	G1002 Technical English					
ti-requisites						
urse Description	The course emphasizes on technical communication at advanced level by exploring critical reading, technical presentation and review writing. The purpose of the course is to enable learners to review literature in any form or any technical article and deliver technical presentations. Extensive activities in practical sessions equip to express themselves in various forms of technical communications. Technical presentations and the module on career setting focus on learners' area of interests and enhance their English language writing skills to communicate effectively.					
urse Out Come	On successful completion of the course the students shall be able to: 1. Develop a critical and informed response reflectively, analytically, discursively, and creatively to their reading. 2. Communicate effectively, creatively, accurately and appropriately in their writing. 3. Deliver technical presentations 4. Design resume and create professional portfolio to find a suitable career					
Course Content: Theory						

odule 1	**tical Reasoning and Writing**	**iting Essays**	**tical Reading**	**4 Classes**
pics: • A Catalog of Reading Strategies • The Myth of Multitasking • A Guide to Writing Essays Speculating about Causes or Effects • Is Google Making Us Stupid (Self Study)				
odule 2	**chnical Presentation**	**esentation**	**al Skills**	**3 Classes**
pics: • Planning the presentation • Creating the presentation • Giving the presentation				
odule 3	**iting Reviews**	**ezi**	**view Writing**	**4 Classes**
Topics: • Review Writing • Short film reviews • Advanced English Grammar (Self Study)				
odule 4	**rting your Career**	**line Writing Lab**	**iting Skills**	**lasses**
pics**:** • Preparing a Resume • Writing Effective Application Letter • Creating a Professional Portfolio				

urse Content: Practical Sessions		
odule 1	**tical Reasoning and Writing**	**lasses**
1. Reading and Analyzing Level 1 – Annotation Level 2 - Assumptions 2. Writing Narrative Essays Level 1 – Draft 1 Level 2 – Draft 2		
Module 2	**Technical Presentation**	**10 Classes**
3. Fishbowl In Fishbowl, students form concentric circles with a small group inside and a larger group outside. Students in the inner circle engage in an in-depth discussion, while students in the outer circle listen and critique content, logic, and group interaction. Level 1 – within group Level 2 – Among 2 group 4. Technical Group Presentation		

Module 3	**Writing Reviews**	**4 Classes**
5. Practice Worksheets Level 1 – Eliminating the Passive Voice Level 2 – Simple, compound and complex sentences 6. Writing Short Film Reviews		
Module 4	**Starting your Career**	**6 Classes**
7. Collaborative Project Job search and writing report Writing Resume		
Module 1-4	**Academic Journal**	**2 Classes**
8. Academic Journal Writing Level 1- Mid Term Level 2 – End Term		
rgeted Application & Tools that can be used: Writing reports, Review writing, Group Discussion, Dyadic interviews, Grammarly.com		
bject work/Assignment:		
Academic Journal – Assignment In Academic Journal (CIJ), students compile task and activities completed in each module and submit to the instructor at the middle and end of the semester.		
ferences 1. Hering, Heik. *How to Write Technical Reports: Understanding Structure, Good Design, Convincing Presentation*. Springer. 2. Johnson, Richard. (2010) *Technical Communication Today.* Pearson, 2015 3. Rice B. Adelrod, Charles R. Cooper and Ellen C. Carillo. (2020) *Reading Critically Writing Well: A Reader and Guide*. Beford/St. Martin's Macmillan Learning, New York. 4. The Princeton Review. (2010) *MCAT Verbal Reasoning & Writing.* The Princeton Review, Inc. 5. https://www.hitbullseye.com/Strong-and-Weak-Arguments.php Accessed on 10 Dec 2021 6. https://www.inc.com/guides/how-to-improve-your-presentation-skills.html Accessed on 10 Dec 2021		
Topics Relevant to "employability": Critical Reasoning, Presentation, Review Writing and Starting Career **Topics Relevant to "Human Values and Professional Ethics":** Critical reasoning		

urse Code: S 1002	**urse Title: Soft skills for Engineers** pe of Course: **Practical Only Course**	P- C	0	2	1
rsion No.	1.0				
urse Pre-requisites	Students are expected to understand Basic English. Students should have desire and enthusiasm to involve, participate and learn.				
ti-requisites	NIL				
urse Description	This course is designed to develop effective communication skills and boost confidence levels. The activity-based modules cover the art of Questioning, how to ask questions, goal setting with emphasis on time and stress management, creating the first impression and introducing one self and finally culminating with the etiquettes of email writing. The pedagogy used will be group discussions, flipped classrooms, continuous feedback, role-play and mentoring.				
urse Objective	The objective of the course is to **familiarize the learners with the concepts of "Soft Skills for Engineers" and attain Skill Development** through **Experiential Learning techniques**.				
urse Out Comes	**On successful completion of this course the students shall be able to:** CO1 **Employ** effective communication skills CO2 **Practice** questioning techniques for better decision making CO3 **Differentiate** individual strengths and weaknesses for self-awareness and stress management CO4 **Recognise** the need to set SMART GOALS				
urse Content:					
odule 1	t of Questioning	le plays		4 classes	
Topics: Note Taking, Framing Open-ended and Close-ended questions, Funnel technique, Probing questions, Leading questions, Rhetorical questions, 5W1H Technique					
	cab Building			Every Class	
Dedicate 5-10minutes towards vocabulary building in every session					

odule 2	al Setting & Time Management	rnal + Outbound training	8 Classes
Goal Setting (SMART Goals), Time Management Matrix, Steps to managing time through outbound group activity, Making a schedule, Daily Plan and calendars (To Do List), Monitoring/charting daily activity			
odule 3	f-introduction and Creating an Impression	ooming checks + Evaluation	8 classes
Topics: Body Language, Grooming guidelines for boys/girls, Common mistakes in Grooming at workplace and social gathering, Etiquettes at work place & social gathering, SWOT – Self-awareness analysis, Self-introduction template, evaluation of self-introduction in class			
odule 4	mail Etiquette	lustry expert / Trainer	4 Classes
Topics: Dos and Don'ts of professional email etiquette, practice writing emails (activity)			
VISION	cap & Summary		2 Classes
Revision of all the modules, overall feedback from the students with regards to the syllabus.			
Targeted Application & Tools that can be used: LMS			
Topics relevant to development of "SKILL": Art of Questioning, Goal Setting & Time Management, Self-introduction and Creating an Impression, E-mail Etiquette for Skill Development through Participative Learning Techniques. This is attained through assessment component mentioned in course handout.			
Catalogue prepared by	L&D Department Faculty members		
Recommended by the Board of Studies on	BOS NO 3 Dated 10 Feb 23		
Date of Approval by the Academic Council	20 ACM dated 15 Feb 23		

urse Code: ECE2010	**urse Title: Innovative Projects using Arduino**	**T-P- C**	-	-			
rsion No.							
urse Pre-requisites	NIL						
ti-requisites	L						
urse Description	This course is designed to provide an in-depth understanding of Arduino microcontrollers and their application in various real time						

	projects involving sensors. Throughout the course, students will learn the fundamentals of Arduino programming and gain hands-on experience with a wide range of sensors. Students will explore how to connect and interface sensors with Arduino boards, read sensor data, and use it to control various output devices This course is suitable for beginners who are interested in exploring the world of electronics and developing practical applications using Arduino and sensors.
urse Objective	The objective of the course is **Employability Skills** of student by using **PARTICIPATIVE LEARNING** techniques.
urse Outcomes	**On successful completion of the course the students shall be able to** 1) Explain the main features of the Arduino prototype board 2) Demonstrate the hardware interfacing of the peripherals to Arduino system. 3) Understand the types of sensors and its functions 4) Demonstrate the functioning of live projects carried out using Arduino system**.**
urse Content:	

odule 1	**Basic concepts of Arduino**	Hands-on	Interfacing Task and Analysis	**4 Sessions**

pics:
Introduction to Arduino, Pin configuration and architecture, Device and platform features, Concept of digital and analog ports, Familiarizing with Arduino Interfacing Board, API's , Introduction to Embedded C and Arduino platform, Arduino Datatypes and variables, Arduino i/o Functions, Arduino Communications, Arduino IDE, Various Cloud Platforms.

odule 2	**nsory Devices**	nds-on	erfacing Task and Analysis	**essions**

duino Sensors: Humidity Sensor, Temperature Sensor, Water Detector / Sensor, PIR Sensor, Ultrasonic Sensor, Connecting Switches and actuators, sensor interface with Arduino.
roduction to 3D Printer: 3D Printer technology and its working Principles, Applications.
Introduction to online Simulators: Working with Tinkercad Simulator.

Topics: Types of Arduino boards, sensors, 3D Printer

rgeted Application & Tools that can be used:

plication Area:

me Automation, Environmental Monitoring, Agriculture and Farming, Industrial

Automation, Internet of Things (IoT), Robotics, Wearable Devices, Security Systems, Education and Learning. These are just a few examples of the many application areas where Arduino and sensors can be applied. The flexibility and affordability of Arduino, combined with the wide range of sensors available, allow for endless possibilities in creating innovative projects. **ofessionally Used Software:** students can use open SOURCE Softwares Arduino IDE and Tincker CAD
oject work/Assignment:
1. Projects: At the end of the course students will be completing the project work on solving many real time issues. **2. Book/Article review: At the end of each module a book reference or an article topic will be given to an individual or a group of students. They need to refer the library resources and write a report on their understanding about the assigned article in appropriate format. Presidency University Library Link .** **Presentation: There will be a presentation from interdisciplinary students group, where the students will be given a project on they have to demonstrate the working and discuss the applications for the same**
xtbook(s): **Monk Simon "Programming Arduino: Getting Started with Sketches", Mc Graw Hill Publications Second Edition**

ferences

ference Book(s)

1. **Neerparaj Rai "Arduino Projects for Engineers" BPB publishers,first edition, 2016.**
2. **Ryan Turner "Arduino Programming " Nelly B.L. International Consulting Ltd. first edition,2019.**

line Resources (e-books, notes, ppts, video lectures etc.):

1. Arduino trending Projects < https://www. https://projecthub.arduino.cc/>
2. Introduction to Arduino < https://onlinecourses.swayam2.ac.in/aic20_sp04/preview>
3. Case studies on Wearable technology< https://www.hticiitm.org/wearables>

ontent:

1. **Cattle Health Monitoring System Using Arduino and IOT (April 2021| IJIRT | Volume 7 Issue 11 | ISSN: 2349-6002)**
2. **M H Hemanth Kumar, Ravi Pratap Singh, Nishu Sharma, Pragya Singh" IOT BASED SMART SECURITY SYSTEM USING ARDUINO" 2021 JETIR August 2021, Volume 8, Issue 8.**
3. **R. Maheswar, P. Jayarajan, S. Vimalraj, G. Sivagnanam, V. Sivasankaran and I. S. Amiri, "Energy Efficient Real Time Environmental Monitoring System Using Buffer Management Protocol," 2018, pp. 1-5, doi: 10.1109/ICCCNT.2018.8494144. https://ieeexplore.ieee.org/document/8494144.**
4. **Yaser S Shaheen,Hussam., " Arduino Mega Based Smart Traffic Control System ," December 2021 Asian Journal of Advanced Research and Reports 15(12): 43-52, 2021(15(12): 43-52, 2021):15(12): 43-52, 2021.**

pics relevant to development of "SKILL": System design for achieving Sustainable Development Goals.

Course Code: MAT1002	**urse Title: Transform Techniques, Partial Differential Equations and Their Applications** **e of Course: School Core**	**T- P- C**			**0**	**3**
rsion No.						
urse Pre-requisites	**AT1001 - Linear Algebra and Calculus**					
ti-requisites	.					
urse Description	This course aims to introduce various transform techniques such as Laplace transform, Fourier transform and Z transform in addition to expressing functions in terms of Fourier series. The course covers applications of Laplace transform to LCR circuits and solution of difference equations using z-transform. The course also deals with the analytical methods for solving partial differential equations and the classical applications of partial differential equations.					
urse Objective	The objective of the course is **Skill Development** of student by using **Problem Solving Techniques.**					
urse Outcomes	On successful completion of this course the students shall be able to: CO-1: Express functions in terms of uniformly convergent Fourier series. CO-2: Apply Laplace transform technique to solve differential equations. CO-3: Employ z-transform technique to solve difference equations. CO-4: Solve a variety of partial differential equations analytically.					
urse Content:						
dule 1	**urier Series**				**10 CLASSES**	
Fourier series: Fourier series - Euler's formulae - Dirichlet's conditions - Change of Interval - half range series – RMS value – Parseval's identity – Computation of harmonics. Engineering Applications of Fourier series.						
dule 2	**egral Transforms**				**15 Classes**	
Laplace Transform: Definition and Laplace transforms of elementary functions. Properties of Laplace transform. Laplace transform of periodic function, unit-step function and impulse function and the related problems. Inverse Laplace transform of standard functions and problems, initial and final value theorems. Convolution theorem, solution of linear ordinary differential equations, LCR circuit problems. **Fourier Transform:** Integral transforms, infinite Fourier transforms, Fourier sine and cosine transforms, inverse Fourier transforms. Engineering Applications of Fourier transform.						
dule 3	**Transform and Difference Equations**				**8 Classes**	
Definition of Z-transform, Z transforms of standard functions and the related problems, standard inverse Z transforms and problems, computation of inverse Z-transform by partial fraction and convolution methods, solution of difference equations using Z-transforms.						

Business and Engineering Applications of Z transform.				
dule 4	**tial Differential Equations**			**12 Classes**

Partial Differential Equations: Formation of PDEs, solution of non-homogeneous PDEs by direct integration, solution of homogeneous PDEs involving derivatives with respect to only one independent variable, method of separation of variables, solution of the Lagrange's PDE of the type Pp + Qq = R.

Applications of PDEs: Various possible solutions of the one dimensional wave and heat equations by the method of separation of variables, D'Alembert's solution of the wave equation, solution of related boundary value problems.

rgeted Applications & Tools that can be used:

plications to electrical engineering, vibrational analysis, acoustics, optics, signal processing, image processing, quantum mechanics, econometrics and shell theory by means of Fourier Series and integral transforms.

ens up new approaches in terms of Z-transform to solving one of the central problems of modern science involving difference equations.

ding the solutions of boundary value problems involving PDEs with reference to wave, heat, and Laplace equations.

signment: Mention the Type of Project /Assignment proposed for this course

Two Assignments based on the applications of the concepts leading to a minimum of 5 engineering problems from a common pool of problems.

xt Book

vin Kreyszig, 2017: " Advanced Engineering Mathematics", 10th Edition, John Wiley.

ferences:

1. B. S. Grewal, 2017: "Higher Engineering Mathematics" 45th Edition, Khanna Publishers.
2. Peter V O'Neil, 2015: "Advanced Engineering Mathematics", 7th Edition, Cengage Learning.
3. Glyn James, 2016: "Advanced Modern Engineering Mathematics", 4th Edition, Pearson Education.
4. Michael D. Greenberg, 2018: "Advanced Engineering Mathematics", 2nd Edition, Pearson Education.

pics relevant to the development of Foundation Skills: All the solution methods.

pics relevant to development of Employability skills: Use of relevant scientific application packages.

urse Code: ECE2011	**urse Title: Innovative Projects using Raspberry Pi**	**T-P- C**	-	-		

rsion No.						
urse Pre-requisites	NIL					
ti-requisites	L					
urse Description	This course is designed to provide an in-depth understanding of Raspberry-pi Single Board Computers and their application in various real time projects involving sensors. Throughout the course, students will learn Raspberry-pi programming and gain hands-on experience with a wide range of sensors. Students will explore how to connect and interface sensors with Raspberry-pi, read sensor data, and use it to control various output devices This course is suitable for advance learners who are interested in exploring the world of electronics and developing practical applications using Raspberry-pi and sensors.					
urse Objective	This course is designed to improve the learners' EMPLOYABILITY SKILLS by using PROBLEM SOLVING Methodologies by using sensors and their interfacing to solve real-time problems **.**					
urse Outcomes	**On successful completion of the course the students shall be able to** 5) Understand the concept of micro python 6) Explain the main features of the Raspberry-pi prototype board 7) Analyse the hardware interfacing of the peripherals to a Single board computer system. 8) Demonstrate the functioning of live projects carried out using Raspberry-pi system					
urse Content:						
dule 1	**Introduction to Micro python**	Hands-on	Interfacing Task and Analysis	**4 Sessions**		
pics: roduction to MicroPython, Comparison with other programming languages, Setting up the MicroPython development environment, Basics of MicroPython syntax and structure.						
dule 2	**rking with Raspberry-pi**	nds-on	erfacing Task and Analysis	**essions**		
roduction to raspberry pi boards, pin-diagram, different types of raspberry pi boards and its application, LED and switch control. Mastering Modules, Setup Raspberry - PuTTY SSH,VNC Viewer to interface with more complicated sensors and actuators. Various Libraries and its functions.						

Topics: Micro Python, types of Raspberry-pi boards, sensors, 3D Printer

rgeted Application & Tools that can be used:

plication Area:

me Automation, Environmental Monitoring, Agriculture and Farming, Industrial Automation, Internet of Things (IoT), Robotics, Wearable Devices, Security Systems, Education and Learning. These are just a few examples of the many application areas where Arduino and sensors can be applied. The flexibility and affordability of Arduino, combined with the wide range of sensors available, allow for endless possibilities in creating innovative projects.

fessionally Used Software: students can use open SOURCE Softwares Thonny Python, Python IDLE etc.

ject work/Assignment:

Projects: At the end of the course students will be completing the project work on solving many real time problems.

2. Book/Article review: At the end of each module a book reference or an article topic will be given to an individual or a group of students. They need to refer the library resources and write a report on their understanding about the assigned article in appropriate format. Presidency University Library Link .

Presentation: There will be a presentation from interdisciplinary students group, where the students will be given a project on they have to demonstrate the working and discuss the applications for the same

xtbook(s):

Monk Simon "Raspberry Pi Cookbook: Software and Hardware Problems and Solutions", Publisher(s): O'Reilly Media, Inc. ISBN: 9781098130923 fourth Edition.

ferences

ference Book(s)

1. Charles Bell Micro Python for the Internet of Things: A Beginner's Guide to Programming with Python on Microcontrollers" by" Edition 1, 2017, ISBN 978-1-4842-3123-4

2. Stewart Watkiss "Learn Electronics with Raspberry Pi " Apress Berkeley, CA . second edition,2020. ISBN978-1-4842-6348-8

line Resources (e-books, notes, ppts, video lectures etc.):

4. Raspberry-pi Projects < https://magpi.raspberrypi.com/articles/category/tutorials/>
5. Introduction to internet of things< https://nptel.ac.in/courses/106105166>
6. Case studies on Wearable technology< https://www.hticiitm.org/wearables>

ontent:

5. Basil, Eliza Sawant, S.D. "IoT based traffic light control system using Raspberry Pi " DOI 10.1109/ICECDS.2017.8389604

6. Supriya S, 2Dr. Aravinda " Green leaf disease detection and identification using

Raspberry Pi https://www.irjet.net/archives/V9/i8/IRJET-V9I847. **7. Dr. E.N. Ganesh., "Health Monitoring System using Raspberry Pi and IOT" DOI : http://dx.doi.org/10.13005/ojcst12.01.03**	
pics relevant to development of "SKILL": System design for achieving Sustainable Development Goals.	
Catalogue prepared by	Dr. Divya Rani /Dr Ashutosh Anand
Recommended by the Board of Studies on	S NO: 17Th BoS meeting held on 5th July 2023
Date of Approval by the Academic Council	ademic Council Meeting No. 21 dated on _____

Course Code: CSE1005	**Course Title:** Programming in Python **Type of Course:** School Core Lab Integrated	**L- T-P-C**	1	0	4	3
Version No.	1.0					
Course Pre-requisites	Basic knowledge of Computers and Mathematics					
Anti-requisites	**NIL**					
Course Description	The purpose of this course is to enable the students to develop python scripts using its basic programming features and also to familiarize the Python IDLE and other software's. This course develops analytical skills to enhance the programming abilities. The associated laboratory provides an opportunity to validate the concepts taught and enhances the ability to build real time applications.					
Course Object	The objective of the course is to familiarize the learners with the concepts of Programming in Python and attain **Employability** through **Problem Solving** Methodologies.					
Course Outcomes	**On successful completion of this course the students shall be able to:** Summarize the basic Concepts of python. 2. Demonstrate proficiency in using data structures. 3. Illustrate user-defined functions and exception handling. 4. Identify the various python libraries.					
Course Content:						
Module 1	Basics of Python programming	Assignment	Programming		**14 Classes**	
Topics: Data types, operators and Expressions, Input and Output Statements. Control Structures – Selective and Repetitive structures						
Module 2	Indexed and	Simple	Programming		**20 Classes**	

	Associative Data Structures	applications		
Topics: Strings, Lists, Sets, Tuples, Dictionaries				
Module 3	Functions, Exception handling and libraries	Case study	Programming	**10 Classes**
Topics: User defined functions, exception handling, Introduction to python built-in libraries				

Targeted Application & Tools that can be used:
Targeted Application : Web application development, AI, Operating systems
Tools: Python IDLE, ANACONDA

Application Areas:
Web Development
Game Development
Scientific and Numeric Applications
Artificial Intelligence and Machine Learning
Software Development
Enterprise-level/Business Applications
Education programs and training courses
Language Development
Operating Systems
Web Scrapping Applications
Image Processing and Graphic Design Applications

Professionally Used Software: Python IDLE, Spyder, Jupyter Notebook, Google Colab

Project work/Assignment:

Project Assignment: Developing python scripts using built in methods and functions

Text Books:

Martin C. Brown, "Python: The Complete Reference", McGraw Hill Education, Forth edition (20 March 2018).
Alex Campbell, "Python for Beginners: Comprehensive Guide to the Basics of Programming, Machine Learning, Data Science and Analysis with Python", August 29, 2021.
Charles Dierbach, "Introduction to Computer Science Using Python", Wiley India Edition,2015.

References:

E. Balagurusamy, "Introduction to Computing and Problem Solving Using Python", Tata McGraw-Hill, 2016
Y. Daniel Liang, "Introduction to Programming Using Python", Pearson, 2017
Brady Ellison, "Python for Beginners: A crash course to learn Python Programming in 1 Week (Programming Languages for Beginners)", August 25, 2021.
Python Tutor - Visualize Python, Java, C, C++, JavaScript, TypeScript, and Ruby code execution
https://practice.geeksforgeeks.org/courses/Python-Foundation

Topics relevant to development of "FOUNDATIONS SKILLS"- Solve the real time problems by analyzing and visualizing the data.
Topics relevant to "HUMAN VALUES & PROFESSIONAL ETHICS"- Data collection and its arrangement

Course Code: CSE3190	**Course Title: Fundamentals of Data Analytics Type of Course: Theory-embedded Lab**	**L-T- P- C**	2	0	2	3
Version No.	3.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	Fundamentals of Data Analytics is designed for inspecting, cleansing, transforming, and modeling data with the goal of discovering useful information, and supports in decision-making. The course begins by covering Data extraction, pre-processing, and transformation. It delivers the basic statistics and taught in an intuitive way to analysis the data. This course will help the students to apply the knowledge on data analysis to a wide range of applications.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Fundamentals of Data Analytics** and attain **SKILL DEVELOPMENT** through **PROBLEM SOLVING** Methodologies.					
Course Out Comes	On successful completion of the course the students shall be able to: 1) **Explain different types of data and variables.** 2) **Interpret data using appropriate statistical methods.** 3) **Demonstrate the collection, processing and analysis of data for any given application and Illustrate various charts using visualization methods.** 4) **Apply the Data Analysis techniques by R Programming**					
Course Content:						
Module 1	Introduction to Data Analysis	Assignment	Data Collection, data analysis, Programming			**8 Sessions**
Topics: Introducing Data, overview of data analysis: Data in the Real World, Data vs. Information, The Many "Vs" of Data, Structured Data and Unstructured Data, Types of Data, Data Analysis Defined, Types of Variables, Central Tendency of Data, Scales of Data, Sources of Data. Data preparation. **R Studio:** Base R-R Studio IDE-Introduction to R Projects and R Markdown. Basic R: R as a Calculator-Scripts and Comments-R Variables. Data I/O: Working Directories-Importing Data Exporting Data-More ways to save-Data I/O in Base R.						
Module 2	Data Analysis and Visualization	Case studies	Programming			**8 Sessions**
Topics: Data Summarization**:** One Quantitative and Categorical Variable. Data Classes: One Dimensional Data Classes-Data Frames and Matrices-Lists. Data Cleaning: Dealing with Missing Data-Strings and Recoding Variables. Manipulating Data in R: Reshaping Data-Merging Datasets. Data Visualizations: Plotting with ggplot2- Plotting with Base R						
Module 3	Statistical Analysis	Case studies	R programming			**7 Sessions**
Topics: Proportion tests-Chi squared test-Fisher exact test-Correlation-T test-Wilcoxon Rank sum tests-Wilcoxon signed rank test- one-way ANOVA test- Kruskal Wallis test						
Module 4	Predictive Analysis	Case studies	Programming			**8 Sessions**

Topics: Linear least-squares – implementation – the goodness of fit – testing a linear model – weighted resampling. Regression using Stats models – multiple regression – nonlinear relationships – logistic regression – estimating parameters – accuracy. Time series analysis – moving averages – missing values – serial correlation – autocorrelation. Introduction to survival analysis

List of Laboratory Tasks:

Experiment No. 1: Introduction to R and RStudio

Level 1: Getting Started with R and RStudio

- Installing R and RStudio.
- Basic R syntax and commands.

Level 2: Working with RStudio

- Understanding the RStudio interface.
- Creating and managing R scripts.

Experiment No. 2: Basic Data Handling in R

Level 1: Data Types and Structures in R

- Vectors, matrices, and data frames.
- Lists and factors.

Level 2: Data Import and Export

- Reading data from CSV, Excel, and text files.
- Exporting data to different formats.

Level 3: Exploring Datasets

- Using functions like head(), summary(), and str().

Experiment No. 3: Basic Data structure in R

Level 1: a. Demonstrate a program to join columns and rows in a data frame using cbind() and rbind() in R.

b.Implement different data structures in R (Vectors, Lists, Data Frames)

Level 2: R AS CALCULATOR APPLICATION a. Using with and without R objects on console

a. Using mathematical functions on console
b. Write an R script, to create R objects for the calculator application

Experiment No. 4: Data Cleaning and Preprocessing

Level 1: Handling Missing Data in R

- Identifying missing values.
- Imputing missing values using mean, median, or other methods.

Level 2: Data Transformation in R

- Standardizing and normalizing data.
- Log-transformations and scaling.

Experiment No. 5: Exploratory Data Analysis (EDA) with R

Level 1: Descriptive Statistics

- Calculating mean, median, and standard deviation.
- Visualizing data using histograms, box plots, and scatter plots.

Experiment No. 6: Data Visualization with ggplot2

Level 1: Demonstrate various graphs that can be made and altered using the ggplot2 package.

Level 2: Create 500 random temperature readings for six cities over a season and then plot the generated data using ggplot2 packages in R

Experiment No. 7: Perform Tests of Hypotheses hypothesis test (parametric)

Level 1: How to perform tests of hypotheses about the mean when the variance is known. How to compute the p-value. Explore the connection between the critical region, the test statistic, and the p-value.

Level 2: A teacher claims that people who work for only five hours per week will score significantly lower than people who work for ten hours per week on a quantitative abilities test. He brings twenty people and randomly assigned them to one or two groups. In one group he has participants who work for ten hours and in another group, he has participants who work for five hours. He conducts the test for all participants. Scores on the test range from one to ten with higher scores representing better performance. Test if there is any significant difference between those who work for five hours per week versus those who work for ten hours per week based on the test performance.

Experiment No 8: Hypothesis – Non-Parametric Test

Level 1: A car manufacturing company like to find the sales of three types of cars produced by them in three regions and is given. Test if there is an association between the regions and types of cars purchased.

Experiment No 9: Correlation and Covariance

Level 1: Using the iris data set in R

a. Find the correlation matrix.
b. Plot the correlation plot on dataset and visualize giving an overview of relationships among data on iris data.
c. Analysis of covariance: variance (ANOVA), if data have categorical variables on iris data.

Level 2 : Ramesh is doing a statistics paper in his post-graduation course. He met his friend Amal who is a textile engineer. Ramesh, who is doing his internship at ABC Researchers, is interested in a question. He poses this question to Amal and tries to find if he can answer. The question is as follows: The data regarding sales of soft- drinks and sales of cotton clothes in a place during the last 12 months are given. Find if there is any association between sales of soft drinks and sales of cotton clothes. Also explain the reason if there is any relationship.

Experiment No 11: Regression Model

Level 1: Import data from web storage (http://www.ats.ucla.edu/stat/data/binary.csv). Name the dataset and now do Logistic Regression to find out the relation between variables that are affecting the admission of a student in an institute based on his or her GRE score, GPA obtained, and rank of the student. Also check the model is fit or not. Require (foreign), require (MASS).

Level 2: Demonstrate multiple regressions, if data have a continuous Independent variable. Apply on the above dataset

Experiment No. 12: Time Series Analysis in R

Level 1: Demonstrate Time series analysis using Time Series Data Library at http://robjhyndman.com/TSDL/.

Targeted Application & Tools that can be used:

Application Area are Decision making in business, health care, financial sector, Medical diagnosis etc.

Text Books

1. Glenn J. Myatt and Wayne P. Johnson, "Making Sense of Data I: A Practical Guide to Exploratory Data Analysis and Data Mining Paperback", Import, 22 July 2014.
2. Introduction to statistics and Data analytics, Christian H, Michael S, Springer,2016
3. Introduction to R- Robert Parker, John Mushcelli and Andrew Jaffe, Johns Hopkins University, 2020 (E-resource)
4. Introduction to Time Series and Forecasting (Springer Texts in Statistics), Peter Brockwell, Richard A. Davis, Springer, 2016.

References

1. Making Sense of Data I: A Practical Guide to Exploratory Data Analysis and Data Mining Paperback, Glenn J. Myatt and Wayne P. Johnson, Import, 22 July 2014.
2. The R Software-Fundamentals of Programming and Statistical Analysis -Pierre Lafaye de Micheaux, Remy Drouilhet, Benoit Liquet, Springer 2013.

Online resources:

http://www.modernstatisticswithr.com/solutions.html#solutionsch3

https://johnmuschelli.com/intro_to_r/ https://users.phhp.ufl.edu/rlp176/Courses/PHC6089/R_notes/
Topics relevant to development of "FOUNDATION SKILLS": 1. Statistical Concepts for data, visualization techniques. 2. Data collection for project based assignments. 3. Inferential Statistics (T test, Z test) 4. Probability Calculation **for Skill Development through Problem Solving methodologies. This is attained through assessment component mentioned in course handout.**

Course Code: CSE2014	**Course Title: Software Engineering** **Type of Course: School Core [Theory Only]**		**L-T- P-C**	**3-0-0-3**
Version No.	1.0			
Course Pre-requisites	**NIL**			
Anti-requisites	**NIL**			
Course Description	The objective of this course is to provide the fundamentals concepts of Software Engineering process and principles. The course covers software requirement engineering processes, system analysis, design, implementation and testing aspects of software system development. The course covers software quality, configuration management and maintenance.			
Course Objectives	The objective of the course is to familiarize the learners with the concepts of Software Engineering and attain Skill Development through Participative Learning techniques.			
Course Out Comes	On successful completion of this course the students shall be able to: 1] Describe the Software Engineering principles, ethics and process models(Knowledge) 2] Identify the requirements, analysis and appropriate design models for a given application(Comprehension) 3] Understand the Agile Principles(Knowledge) 4] Apply an appropriate planning, scheduling, evaluation and maintenance principles involved in software(Application)			
Module 1	**Introduction to Software Engineering and Process Models (Knowledge level)**	Quiz		**09 Hours**
Introduction: Need for Software Engineering, Professional Software Development, Software Engineering Ethics, Software Engineering Practice-Essence of Practice, General Principles Software Development Life Cycle **Models:** Waterfall Model – Classical Waterfall Model, Iterative Waterfall Model, Evolutionary model-Spiral, Prototype.				
Module 2	**Software Requirements,**	Assignment	Development of SRS	**11 Hours**

	Analysis and Design (Comprehension level)		documents for a given scenario	
Requirements Engineering: Eliciting requirements, Functional and non- Functional requirements, Software Requirements Specification (SRS), Requirement Analysis and validation. Requirements modelling- Introduction to Use Cases, Activity diagram and Swim lane diagram. CASE support in Software Life Cycle, Characteristics of CASE Tools, Architecture of a CASE Environment. **Design:** Design concepts, Architectural design, Component based design, User interface design.				
Module 3	**Agile Principles & Devops (Knowledge level)**	Quiz		**09 Hours**
Agile: Scrum Roles and activities, Sprint Agile software development methods - Scaling, User Stories, Agile estimation techniques, Product backlogs, Stake holder roles, Dynamic System Development Method. **Devops:** Introduction, definition, history, tools.				
Module 4	**Software Testing and Maintenance (Application Level)**	Assignment	Apply the testing concepts using Programing	**12 Hours**
Software Testing-verification and validation, Test Strategies - White Box Testing, Black box Testing. Automation Tools for Testing. **Software Quality Assurance**-Elements of software quality assurance, SQA Tasks, Goals and Metrics, Software configuration management- SCM process, SCM Tools (GitHub). **Maintenance-** Characteristics of Software Maintenance, Software Reverse Engineering, Software Maintenance Process Models.				
Targeted Application & Tools that can be used: Selenium, GitHub, CASE Tools				
Text Book oger S. Pressman, "Software Engineering – A Practitioner's Approach", VII Edition, McGraw-Hill, 2017. ob Hughes, Mike Cotterell, Rajib Mall, "Software Project Management", VI Edition, McGraw-Hill, 8.				
References b Mall, "Fundamentals of Software Engineering", VI Edition, PHI learning private limited, 2015. Sommerville, "Software Engineering", IX Edition, Pearson Education Asia, 2011. e Software Development Principles, Patterns and Practices.1st Edition, Wiley, 2002				
Topics Relevant to "Skill Development: Balck box Testing, White box Testing, Automated Testing for Skill development through Participative Learning Techniques. This is attained through assessment mentioned in the course handout				

Course Code:CSE 2066	**Course Title:** Computer Graphics	**L-P-C**	3	0	0		3
Version No.	2.0						
Course Pre-requisites	C Programming						
Anti-requisites	NIL						

Course Description	This course demonstrates the basics of graphics and visualization in computer science, enabling students to appreciate how the computer system displays graphics and visual effects on a display device. The course uses assignments to develop visualization skills of the students. The key topics covered in this course include algorithms for drawing basic primitives, transformations, viewing and clipping for both 2D and 3D objects along with Bezier curves and Surfaces.		
Course Objective	The objective of the course is to familiarize the learners with the concepts of **ComputerGraphics and** attain **Skill Development** through **Participative Learning** techniques.		
Course Out Comes	On successful completion of the course the students shall be able to: CO 1: Illustrate algorithms for drawing basic primitives like Point, Line andPolygon. CO 2: Illustrate algorithms for performing 2D Geometric Transformations, viewing and clipping. CO 3: Illustrate algorithms for performing 3D Geometric Transformations, clipping.CO 4: Describe plane Bezier curves and Bezier surfaces.		
Course Content:			
Module 1	**Overview: Basics of Computer Graphics**	Assignment	**No. of Sessions 13**
Topics: An Introduction Graphics System: Computer Graphics and Its Types, Application of computer graphics. Graphics Systems: Video Display Devices, Raster Scan Systems, Random Scan Systems, Raster graphics Vs. Random Graphics, Flat panel Displays – emissive and non-emissive displays, Input Devices, logical inputs, Graphics tools andsoftware Line drawing algorithms - Midpoint, DDA, Bresenham's. Circle generation algorithms - Midpoint circle drawing algorithm, Bresenham's circle algorithm. Basics of 2D and 3D objects. **Assignment:** Numerical problems based on Line and circle drawing algorithm			
Module 2	**2D Geometric Transformations, viewing andclipping**	Assignment	**No. of Sessions : 12**
2D Geometric Transformations: Basics of translation, scaling, rotation, reflection and shearing. Matrix representations and homogeneous coordinates for translation, scaling, rotation, reflection and shearing. 2D Composite transformations, General pivot point rotation and scaling. Introduction to OpenGL concepts and libraries. OpenGL geometric transformations functions. Basics of 2D viewing and Clipping: Basics of viewing and Clipping, 2D viewing pipeline, Viewing Transformation systems, Normalization and Viewport Transformation Types of clipping: point, Line and polygon clipping, 2D line clipping algorithms: cohen-sutherland line clipping, Liang-Barsky line clipping algorithm, polygon fill area clipping: Sutherland-Hodgeman polygon clipping algorithm,OpenGL 2D viewing and clipping functions. **Assignment:** Numerical problems based on 2D transformations.			
Module 3	**3D Geometric Transformations,clipping:**	Mini-project	**No. of Sessions : 11**
3D Geometric Transformations: 3D translation, rotation, reflection and shearing, composite 3D scaling, transformations, OpenGL 3D geometric transformations functions, Transformations between 3D CoordinateSystems. Basics of 3D Viewing and Clipping: 3D viewing concepts, 3D viewing coordinate parameters, Transformation from world to viewing coordinates, Projection transformation, parallel projections - orthogonal projections and obliqueprojections, parallel-Projection Transformation Matrix, perspective projections, Perspective-Projection Transformation Matrix Assignment: Based on the activities in the link: pu.informatics.global			

Module 4	**Plane curves and surfaces**	Quiz	**No. of Classes : 9**

Plane Curves: Plane Curves representation, Nonparametric Curves, Parametric Curves, Curved Surfaces, QuadricSurfaces.

Basics of Curves and surfaces: Interpolation and Approximation Splines, Parametric Continuity Conditions, Geometric Continuity Conditions, Spline Specifications. Representation of Space Curves, Cubic Splines, Bezier Curves, Parametric Cubic Curves, Quadric Surfaces, Bezier Surfaces. OpenGL Quadric-Surface and Cubic-Surface Functions

Targeted Application & Tools that can be used:Application Area: Game design and Animation

Tools/Simulator/Software used: Visual Studio 17.0 / CodeBlock

Text Book:

T1: Donald D. Hearn, M. Pauline Baker and Warren Carither, Computer Graphics with OpenGL, Pearson Education, 4th Edition, 2021

Reference Books:

R1. John F Hughes, Andries van Dam, Steven K. Feiner, James D. Foley, Morga, Computer Graphics: Principlesand Practice, Pearson Education India, Third Edition, 2013

R2. John Kessenich, Graham Sellers, Dave Shreiner , OpenGL Programming guide , Addison-Wesley Ninth Edition,2016

R3. Edward Angel and Dave shreiner, Interactive Computer Graphics, A top down approach with shader basedOpenGL, Pearson Education, 6th Edition, 2018

E-References

https://presiuniv.knimbus.com/user#/home

Topics relevant to development of "Skill Development":

1. Line drawing algorithms (DDA, Bresenham's)
2. Graphics tools and software
3. Liang-Barsky line clipping algorithm
4. cohen-sutherland line clipping
5. OpenGL 2D viewing and clipping functions

for Skill Development through Participative Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE3156	**Course Title: Database Management Systems** **Type of Course: 1) School Core** **2) Laboratory Integrated**	**L-T-P-C**	3	0	2	4
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					

Course Description	This course introduces the core principles and techniques required in the design and implementation of database systems. It covers concepts of relational database systems (RDBMS). More emphasis is set on how to design, develop, organize, maintain and retrieve information efficiently. It helps the students to learn and practice data modeling and database designs. The course also introduces the concept of object oriented and object relational databases. The associated laboratory is designed to implement database design using MySQL DATABASE in information technology applications. All the exercises will focus on the fundamentals for creating, populating, sophisticated, interactive way of querying, and simultaneous execution of the transactions of database.			
Course Objective	The objective of the course is to familiarize the learners with the concepts of Database Management Systems and attain **Employability** through **Problem Solving** Methodologies.			
Course Out Comes	On successful completion of the course the students shall be able to: 1] Demonstrate a database system using ER model and relational algebra. [Understanding] 2] Build databases using SQL queries query processing. [Applying] 3] Apply the functional dependencies and design the database using normalization. [Applying] 4] Interpret the concept of object-oriented databases and object-relational databases. [Understanding]			
Course Content:				
Module 1	**Introduction to Database Modelling and Relational Algebra (Understanding)**	Assignment	**Problem Solving**	**8 Classes**
Topics: **Introduction to Database:** Schema, Instance, 3-shema architecture, physical and logical data independence, Data isolation problem in traditional file system, advantages of database over traditional file systems. Entity Relationship (ER) Model, ER Model to Relational Model, Examples on ER model. **Relational Algebra** with selection, projection, rename, set operations, Cartesian product, joins (inner and outer joins), and division operator. Examples on Relational Algebra Operations.				
Module 2	**Fundamentals of SQL and Query Optimization (Applying)**	Assignment	**Programming**	**8 Classes**
Topics: **SQL Database Querying,** DDL, DML, Constraints, Operators, Set Operators, Aggregate Functions, Joins, Views, Procedures, Functions and Triggers. **Database programming issues and techniques:** Embedded SQL, Dynamic SQL; SQL / PSM and NoSQL. **Query Optimization:** Purpose, transformation of relational expressions, estimating cost and statistics of expression, choosing evaluation plans, linear and bushy plans, dynamic programming algorithms.				
Module 3	**Relational Database Design & Transaction Management (Applying)**	Assignment	**Problem Solving**	**12 Classes**

Topics:
Relational database design: Problems in schema design, redundancy and anomalies, Normal Forms based on Primary Keys-(1NF,2NF, 3NF), Boyce-Codd Normal Form, Multi valued Dependency (Fourth Normal Form), Join Dependencies (Fifth Normal Form), lossy and lossless decompositions, Database De-normalization.
Transaction Management: The ACID Properties; Transactions and Schedules; Concurrent Execution of Transactions; Lock- Based Concurrency Control; Performance of locking; Transaction support in SQL; Introduction to crash recovery; 2PL, Serializability and Recoverability; Lock Management; The write-ahead log protocol; Check pointing; Recovering from a System Crash; Media Recovery; Other approaches and interaction with concurrency control.

Module 4	**Advanced DBMS Topics (Understanding)**	Assignment	**Case Study**	**8 Classes**

Topics:
Advanced topics: Object oriented database management systems, Deductive database management systems, Spatial database management systems, Temporal database management systems, Constraint database management systems.
New database applications and architectures such as Data warehousing, Multimedia, Mobility, NoSQL, Native XML databases (NXD), Document-oriented databases, Statistical databases.

List of Laboratory Tasks:
Create Employee, Student, Banking and Library databases and populate them with required data. Do the following experiments of different lab sheets on those databases.

Labsheet-1 [3 Practical Sessions]
Experiment No 1: [1 Session]

1. **To study and implement the different language of Structured Query Language.**

Level 1: Perform operations using Data Definition Language and Data Manipulation Language commands including different variants of SELECT on Student DB.
Level 2: Identify the given requirements; valid attributes and data types and Perform DDL and DML operations on a given scenario. [Banking Databases]
Experiment No. 2: [2 Sessions]

2. **To study and implement the concept of integrity constraints in SQL.**

Level 1: Create tables on Banking database using PRIMARY KEY, NOT NULL, UNIQUE, FOREIGN KEY and demonstrate the working of relational, logical, pattern matching, BETWEEN, IS NULL, IN and NOT IN Special Operators on Student Database.
Level 2: Enforce different types of data and referential integrity constraints. Then try queries with special operators based on the student database. [Banking Database].

Labsheet-2 [3 Practical Sessions]
Experiment No. 3: [1 Session]

3. **Implement complex queries in SQL.**

Level 1: Implement the conjugate of GROUP BY, ORDER BY and aggregate functions on Banking Database**.**
Level 2: Implement MySQL DB queries on library database using appropriate clauses and aggregate functions**.** Also order the data either in ascending and descending order using corresponding clause. [Library databases].

Experiment No. 4: [2 Session]

4. **To study and implement different types of Set and Join Operations [2 Slots]**

Level 1: Demonstrate different types of Set Operations (UNION, UNION ALL, INTERSECT, MINUS) and Join Operations (INNER JOINs, OUTER JOINS, CROSS JOIN, NATURAL JOIN) on two or more tables of Airline Database.
Level 2: Use Set and Join operations to retrieve the data from two or more relations(tables) as per the given scenario. [Airline Database]

Labsheet-3 [2 Practical Sessions]
Experiment No. 5: [2 sessions]

5. To study and implement Views, and Procedures in MySQL DB.

Level 1: Implement MySQL Views, and Procedures in ORACLE DB on Employee database.
Level 2: Analyze the requirement and construct views, and Procedures on Mini Project Domain. [Banking Database]

Labsheet-4 [2 Practical Sessions]
Experiment No. 6: [2 Sessions]

6. To study and implement Functions, and Triggers in MySQL DB.

Level 1: Implement Oracle Functions and Triggers in Oracle on Employee database.
Level 2: Analyze the requirement and construct Functions and Triggers. [Supply chain Database]

Labsheet-5 [2 Practical Sessions]
Experiment No. 7: [2 Sessions]
To implement the concept of forms and reports.
Level 1: Implement the concept of forms and reports.
Level 2: Analyze the schema relationship.

Labsheet-6 [2 Practical Sessions]
Experiment No. 8: [2 Sessions]
Design a mini project based on the databases such as Inventory Management System, University Management System, Hospital Management System, etc.
Level 1: Implement the real time database.
Level 2: Analyze the working of database in real time.

Targeted Application & Tools that can be used:
Application Area: Relational database systems for Business, Scientific and Engineering Applications.
Tools/Simulator used: MySQL DB for student practice.
Also demonstration of ORACLE DB on object-relational database creation and JDBC connection.

Percentage of changes in this version: 50% of changes from earlier version. New topics are highlighted initalic.

1. Problem Solving: Constructing ER-Diagrams for a given real time requirements, Normalizing the databases, querying the databases using relational algebra.
2. Programming: Implementation of any given scenario using MySQL.

Text Book

1] RamaKrishna & Gehrke, "Database Management Systems" 3rd Edition, 2018, McGraw-Hill Education.
2] Avi Silberschatz, Henry F. Korth, S. Sudarshan, "Database System Concepts", McGraw-Hill ,7th Edition, 2019.
3] W. Lemahieu, S. vanden Broucke and B. Baesens, "Principles of Database Management: Practical Guide to Storing, Managing and Analyzing Big and Small Data", Cambridge University Press, 2018.

References

1] Elmasri R and Navathe S B, "Fundamentals of Database System", Pearson Publication, 7th Edition, 2018.
2] M. Kleppmann, "Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems", O'Reilly, 2017.

Topics relevant to development of "FOUNDATION SKILLS": S - Skill Development: Relational database design using ER- Relational mapping, Implementation of given database scenario using MYSQLDB.
Topics relevant to development of Employability: Develop, test and implement computer databases, creating sophisticated, interactive and secure database applications
Topics relevant to "HUMAN VALUES &PROFESSIONAL ETHICS": Nil

Course Code: PPS4002	**Course Title: Introduction to Aptitude Type of Course: Practical Only Course**	**L- P- C**	0	2	1
Version No.	1.0				
Course Pre-requisites	Students should know the basic Mathematics & aptitude along with understanding of English				
Anti-requisites	Nil				
Course Description	The objective of this course is to prepare the trainees to tackle the questions on various topics and various difficulty levels based on Quantitative Ability, and Logical Reasoning asked during the placement drives. There will be sufficient focus on building the fundamentals of all the topics, as well as on solving the higher order thinking questions. The focus of this course is to teach the students to not only get to the correct answers, but to get there faster than ever before, which will improve their employability factor.				
Course Objective	The objective of the course is to familiarize the learners with the concepts of Aptitude and attain Skill Development through Problem Solving techniques.				
Course Outcomes	On successful completion of the course the students shall be able to: CO1] **Recall** all the basic mathematical concepts they learnt in high school. CO2] **Identify** the principle concept needed in a question. CO3] **Solve** the quantitative and logical ability questions with the appropriate concept. CO4] **Analyze** the data given in complex problems. CO5] **Rearrange** the information to simplify the question				
Course Content:					
Module 1	Quantitative Ability	Assignment	Bloom's Level : Application		**02 Hours**
Topics: Introduction to Aptitude, working of Tables, Squares, Cubes					
Module 2	Logical Reasoning	Assignment	Bloom's Level : Application		**18 Hours**
Topics: Linear & Circular Arrangement Puzzle, Coding & Decoding, Blood Relations, Directions, Ordering and Ranking, Clocks and Calendars, Number Series, Wrong number series, Visual Reasoning					
Targeted Application & Tools that can be used: Application area: Placement activities and Competitive examinations. Tools: LMS					

Text Book Quantitative Aptitude by R S Aggarwal Verbal & Non-Verbal Reasoning by R S Aggarwal
References www.indiabix.com www.youtube.com/c/TheAptitudeGuy/videos
Topics relevant to Skill development: Quantitative and reasoning aptitude for Skill Development through Problem solving Techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE2001	**Course Title:** Data Structures and Algorithms **Type of Course: Integrated**		**L- T-P- C**	3-0-2-4
Version No.	1.0			
Course Pre-requisites	Problem Solving Using Java			
Anti-requisites	NIL			
Course Description	This course introduces the fundamental concepts of data structures and to emphasize the importance of choosing an appropriate data structure and technique for program development. This course has theory and lab component which emphasizes on understanding the implementation and applications of data structures using Java programming language. With a good knowledge in the fundamental concepts of data structures and practical experience in implementing them, the student can be an effective designer, developer for new software applications.			
Course Objective	The objective of the course is to familiarize the learners with the concepts of Data Structures and Algorithms and attain Skill Development through Experiential Learning techniques.			
Course Out C omes	**On successful completion of the course the students shall be able to:** **CO1:** Implement program for given problems using fundamentals of data structures. [Application] **CO2:** Apply an appropriate linear data structure for a given scenarios. [Application] **CO3:** Apply an appropriate non-linear data structure for a given scenarios. [Application] **CO4:** Explain the performance analysis of given searching and sorting algorithms.			
Course Content:				
Module 1	**Introduction to Data Structure and Linear Data Structure –**	Assignment	Program activity	**18 Sessions**

	Stacks and Queues			
Introduction – Introduction to Data Structures, Types and concept of Arrays. **Stack -** Concepts and representation, Stack operations, stack implementation using array and Applications of Stack. **Queues -** Representation of queue, Queue Operations, Queue implementation using array, Types of Queue and Applications of Queue.				
Module 2	**Linear Data Structure- Linked List**	Assignment	Program activity	**17 Sessions**
Topics: Linked List - Singly Linked List, Operation on linear list using singly linked storage structures, Circular List, Applications of Linked list. **Recursion** - Recursive Definition and Processes, Programming examples.				
Module 3	**Non-linear Data Structures - Trees and Graph**	Assignment	Program activity	**15 Sessions**
Topics: Trees - Introduction to Trees, Binary tree: Terminology and Properties, Use of Doubly Linked List, Binary tree traversals: Pre-Order traversal, In-Order traversal, Post - Order traversal. **Graph** - Basic Concept of Graph Theory and its Properties, Representation of Graphs.				
Module 4	**Searching & Sorting Performance Analysis**	Assignment	Program activity	**14sessions**
Topic: Sorting & Searching - Sequential and Binary Search, Sorting – Selection and Insertion sort. **Performance Analysis** - Time and space analysis of algorithms – Average, best and worst case analysis.				
List of Laboratory Tasks: **Lab sheet -1** **Level 1:** Prompt the user, read input and print messages. Programs using class, methods and objects **Level 2:** Programming Exercises on fundamental Data structure - Arrays based on Scenario. **Lab sheet -2** **Level 1:** Programming Exercises on Stack and its operations **Level 2:** Programming Exercises on Stack and its operations with condition **Lab sheet -3** **Level 1:** Programming on Stack application infix to postfix Conversion **Level 2:** - **Lab sheet -4** **Level 1:** Programming Exercises on Queues and its operations with conditions **Level 2:** - **Lab sheet -5** **Level 1:** Programming Exercises on Linked list and its operations. **Level 2:** Programming Exercises on Linked list and its operations with various positions **Lab sheet -6** **Level 1:** - **Level 2:** Programming scenario based application using Linked List				

Lab sheet -7

Level 1: Programming Exercises on factorial of a number

Level 2: Programming the tower of Hanoi using recursion

Lab sheet -8

Level 1: -

Level 2: Programming the tower of Hanoi using recursion

Lab sheet -9

Level 1: Programming Exercise on Doubly linked list and its operations

Level 2: -

Lab sheet -10

Level 1: Program to Construct Binary Search Tree and Graph

Level 2: Program to traverse the Binary Search Tree in three ways(in-order, pre-order and post-order) and implement BFS and DFS

Lab sheet -11

Level 1: Program to Implement the Linear Search & Binary Search

Level 2: Program to Estimate the Time complexity of Linear Search

Lab sheet -12

Level 1: Program to Implement and Estimate the Time complexity of Insertion Sort

Level 2: Program to Implement and Estimate the Time complexity of Insertion Sort

Lab sheet -13

Level 1: Program to Implement and Estimate the Time complexity of Selection Sort

Level 2: Program to Implement and Estimate the Time complexity of Selection Sort

Targeted Application & Tools that can be used

Use of PowerPoint software for lecture slides and use of Ubuntu for lab programs to execute. Tool is Codetantra tool.

Project work/Assignment:

Assignment: Students should complete the lab programs by end of each practical session and module wise assignments before the deadline.

Text Book

T1 Narasimha Karumanchi: *"Data Structures and Algorithms Made Easy in Java",* 5th Edition, CareerMonk Publications, 2017.

References

R1 Mark Allen Weiss: *"Data Structures and Algorithm Analysis in Java",* 4th Edition, Pearson Educational Limited, 2014.

R2 Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser: *"Data Structures and Algorithms in Java"*, 6th Edition, John Wiley & Sons, Inc., ISBN: 978-1-118-77133-4, 2014.

R3 Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, 2017: *"Introduction to Algorithms",* 3rd Edition, PHI Learning Private Limited.

Web resources:

For theory: https://onlinecourses.nptel.ac.in/noc20_cs85/preview For Lab : codetantra tool https://puniversity.informaticsglobal.com/login

Topics relevant to "SKILL DEVELOPMENT": Linked list and its type, Tree traversal and hashing tables for Skill Development through Experiential Learning techniques. This is attained through the assessment component mentioned in the course handout.

urse Code: AT2003	**Course Title: NUMERICAL METHODS FOR ENGINEERS pe of Course: School Core**	**-T- P-C**	1	0	2	2
rsion No.						
urse Pre-requisites	AT1002 – Transform Techniques, Partial Differential Equations and Their Applications					
ti-requisites						
urse Description	The course focuses on formulating and solving problems concerning real-world engineering applications numerically as well as statistically. This course provides an introduction to basic numerical methods to deal with algebraic and transcendental equations, system of equations, interpolation, differentiation and integration. This course also deals with numerical solution of ordinary differential equations by means of Taylor's series method, modified Euler's method and Runge-Kutta methods.					
urse Objective	The objective of the course is to **familiarize the learners with the concepts of " NUMERICAL METHODS FOR ENGINEERS" and attain <u>Skill Development</u> Through <u>Problem Solving.</u>**					
urse Outcomes	On successful completion of the course the students shall be able to: 1] Solve algebraic and transcendental equations numerically. 2] Adopt numerical techniques to differentiate and integrate functions. 3] Apply numerical methods to solve ordinary differential equations.					
urse Content:						
dule 1	**merical solution of Algebraic and Transcendental Equations**					**15 Classes**
Algebraic and Transcendental Equations, Regula - Falsi method, Bisection method (Self study), Secant method, Newton-Raphson method, and NR method for non-linear Equations, Fixed-point iteration method. System of Linear Equations: Introduction, LU decomposition method, Gauss-Jacobi method, Gauss-Seidel iteration method, Largest Eigen value and corresponding Eigen vector by Power method & Jacobi Method.						
dule 2	**merical Interpolation, differentiation and Integration**					**15 Classes**
Numerical Interpolation: Newton's forward and backward interpolation method, Newton's divided difference method, Lagrange's method, numerical differentiation. Numerical integration: Trapezoidal rule, Simpson's one-third rule, Simpson's three-eighth rule, Weddle's Rule. Area between the two curves.						
dule 3	**merical solution of**					**15 Classes**

	ODEs and PDEs			

Solution of ordinary differential equations: Initial Value problems: Taylor's series method, Picard's method, Euler's Method, Modified Euler's method, Runge-Kutta method, Milne's predictor-corrector formula. Adams -Bashforth method**,** Boundary value problems - Finite difference methods for ODE. Numerical solution for LCR & damped forced oscillatory equations.

Solution of partial differential equations: Schmidt Explicit Formula for Heat Equation, Crank-Nicolson method. Numerical solution to Wave, Laplace & Heat Equation.

rgeted Application & Tools that can be used:

The objective of the course is to familiarize students with a variety of numerical techniques and the theoretical concepts of probability and statistics so as to equip them with the necessary numerical approaches and basic statistical tools to tackle engineering and real-life problems.

signment:

Gauss-Jacobi iteration method.
Numerical differentiation.
Gaussian quadrature rule for numerical integration.
Taylor series method for ODEs.
Implicit and explicit schemes for PDEs.

xt Books

T1: M.K. Jain, S.R.K. Iyengar and R.K. Jain, Numerical Methods for Scientific and Engineering Computations, 6th Edition, New age Publishing House, 2015.

T2: Erwin Kreyszig, "Advanced Engineering Mathematics", 10th Edition, John Wiley& Sons (India), 2014.

ferences:

R1: B.S. Grewal, Numerical methods in engineering and science, 10th Edition, Khanna publishers, 2016.

R2: B.S. Grewal, "Higher Engineering Mathematics", 44th edition, Khanna Publishers.

R3: Steven C Chapra and Raymond P Canale, "Numerical Methods for Engineers," 7th Ed., McGraw-Hill Edition, 2015.

R4: C. Ray Wylie and Louis C Barrett, "Advanced Engineering Mathematics", 6th Edition, McGraw-Hill, 2012.

Topics relevant to SKILL DEVELOPMENT: This course focuses on formulating and solving problems concerning real-world engineering applications numerically as well as statistically. This course provides an introduction to basic numerical methods to deal with algebraic and transcendental equations, system of equations, interpolation, differentiation and integration with numerical solution of ordinary differential equations by means of Taylor's series method, modified Euler's method and Runge-Kutta methods for **Skill Development through Problem Solving methodologies.** This is attained through assessment component mentioned in course handout.

urse Code: E2007	**urse Title: Design and Analysis of Algorithms** **pe of Course: Program Core & Theory only**	**-P- C**	3			3

rsion No.		
urse Pre-requisites		CSE2001, Data Structure and Algorithms
ti-requisites		
urse Description		This intermediate course enables students to design and analyze efficient algorithms to solve problems. This course covers typical design methods such as divide-and-conquer, dynamic programming and greedy method to solve problems. The students shall develop strong analytical skills as part of this course.
urse Objectives		This course is designed to improve the learners' EMPLOYABILITY SKILLS by using PROBLEM SOLVING Methodologies.
urse Outcomes		**On successful completion of the course the students shall be able to:** **1] Identify the efficiency of a given algorithm. [Comprehension]** **2] Employ divide and conquer approach to solve a problem. [Application]** **3] Illustrate dynamic programming approach to solve a given problem. [Application]** **4] Solve a problem using the greedy method. [Application]** **5] Discuss the techniques to solve a real-world problem based on its complexity classes. [Comprehension]**
urse Content:		

dule 1	Introduction to Algorithms	ignment		Problem Solving	**Sessions**
	pics: Algorithm Design and efficiency, measuring of running time of algorithms. Insertion sort and merge sort, Asymptotic Growth and Notations. Recurrences--Masters method. **Assignment:** Comparatively evaluate bubble sort, insertion sort and mergesort.				
dule 2	Review of Searching and Sorting techniques	ignment		Programming/ Problem Solving	**Sessions**
	pics: **Divide and Conquer:** Examples. Strassen's Matrix multiplication. **Sorting:** Quicksort, Heapsort, Lower bound of comparison-based sorting, non-comparison-based sorting: Radix sort. **Search:** Review of Linear Search and Binary Search, Hashing and hash tables. **ignment:** Design and develop an algorithm using Divide and Conquer technique for a given scenario.				
dule 3	Greedy Algorithms	ignment		Programming/ Problem Solving	**Sessions**

pics:

roduction, Fractional Knapsack Problem, Minimal Spanning Tree: Prim's Algorithm and Kruskal's Algorithm, Single-source Shortest Path: Dijkstra's Algorithm. Huffman Codes.

ignment: Design and Develop a solution to a given scenario using greedy method.

dule 4	namic Programming	ignment		Programming/ Problem Solving	**Sessions**

Topics:

Introduction with examples, Principles of Memoization, 0-1 Knapsack Problem, Bellman-Ford algorithm, Floyd-Warshall's Algorithms. Optimal Binary Search Trees, Chain Matrix Multiplication.

Assignment: For a given scenario, attempt the three design paradigms learned so far and argue the best approach to solve the problem

dule 5	Complexity Classes and Heuristics	Assignment		Programming/ Problem Solving	**09 Hours**

pics:

plexity classes: P, NP, and NP-Complete Problems. Backtracking: n-Queens. Branch and bound: Travelling Salesman Problem.

Assignment: Apply backtracking algorithmic designing technique for solving queen's problems for 4, 8 and 16 inputs.

geted Application & Tools that can be used:

plication Area is to Design and Analyzing the efficiency of Algorithms. This fundamental course is used by all application developers.

fessionally Used Software: GCC compiler.

ject work/Assignment:

1. **Problem Solving: Design of Algorithms and implementation of programs.**
2. **Programming: Implementation of given scenario using Java.**

Text Book:

T1. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, *'Introduction to Algorithms'*, MIT Press, 2022.

T2. J. Kleinberg and E. Tardos, *'Algorithm Design'*, Addison-Wesley, 2005.

ferences

R1. Anany Levitin, *'Introduction to the Design and Analysis of Algorithms'*, Pearson Education, 2003.

R2. Tim Roughgarden, *'Algorithms Illuminated'* (books 1 through 3), Soundlikeyourself Publishing, 2017,18,19 respectively.

R3. AV Aho, J Hopcroft, JD Ullman, *'The Design and Analysis of Algorithms'*, Addison-Wesley, 1974.

urse Code: AT2004	**urse Title: Discrete Mathematical Structures** **e of Course: Program Core**	**- P- C**	3	0	0	3
rsion No.						
urse Pre-requisites						
ti-requisites						
urse Description	The course provides insights into the fundamental aspects of mathematical logic and predicate calculus. The course delves deeply into the concepts of algebraic structures, lattices and Boolean algebras which are widely used in computer science and engineering. It also highlights the principles of counting techniques and their applications.					
urse Objective	The objective of the course is **Skill Development** of student by using **Problem Solving Techniques.**					
urse Outcomes	On successful completion of the course the students shall be able to: CO1: Explain logical sentences through predicates, quantifiers and logical connectives. CO2: Comprehend the basic principles of set theory and different types of relations. CO3: Elucidate the concepts of lattices and Boolean algebra. CO4: Deploy the counting techniques to tackle combinatorial problems.					
urse Content:						
dule 1	**Mathematical Logic and Predicate Calculus**			**2 classes**		
Propositional Logic, Propositional Logic Equivalences, Normal forms, Inference rules, Introduction to Proofs, Conversion to clausal form, Predicate calculus, The Statement function, Inference theory of the Predicate Calculus.						
dule 2	**ebraic Structures**			**0 classes**		
Sets and set-operations, functions, relations and their properties & representations of relation by matrix, closure of different type of relations, equivalence relations, primitive recursive function.						
dule 3	**tices and Boolean Algebra**			**1 classes**		
Partial ordering, Posset, Lattices & Algebraic structures, Sub lattice, Basic properties of algebraic systems by lattices, Distributive lattices, complement of an element in a lattice, Boolean lattice & Boolean algebra, cancellation laws and unique complement theorem.						
dule 4	**nciples of Counting Techniques**			**2 classes**		
Chinese Remainder Theorem, pigeonhole principle, generalized pigeonhole principle, Generalized Permutations and Combinations, Recurrence Relations.						

rgeted Application & Tools that can be used:

crete mathematics provides the mathematical foundations for many computer science courses including data structures, algorithms, database theory, automata theory, formal languages, compiler theory, computer security, and operating systems.

oject work/Assignment: Mention the Type of Project /Assignment proposed for this course

ignment 1: Logic Equivalences and Predicate calculus.
ignment 2: Equivalence Relations and Lattices
ignment 3: Recurrence Relations

xt Books

Kenneth H. Rosen, "Discrete Mathematics and its Applications", McGraw-Hill's 7th Edition, 2011.

Kolman, Bernard; Busby, Robert C; Ross, Sharon Cutler," Discrete mathematical structures", Pearson India, 6th Edition, 2015.

, C L Mohapatra, D P.," Elements of Discrete Mathematics a Computer oriented approach", New Delhi McGraw Hill Education, 4th Edition, 2015.

ott, Joe L; Kandel, Abraham; Baker, Theodore P, "Discrete Mathematics for Computer Scientists and Mathematicians", Pearson India, 2nd Edition, 2015.

o, Susanna S, "Discrete Mathematics with applications", New Delhi Cengage Learing, 4th Edition, 2016.

ferences:

Tremblay, J.P. and Manohar.R, "Discrete Mathematical Structures with Applications to Computer Science", Tata McGraw Hill Pub. Co. Ltd, New Delhi, 30th Reprint, 2011.

Grimaldi, R.P. "Discrete and Combinatorial Mathematics: An Applied Introduction", 4th Edition, Pearson Education Asia, Delhi, 2007.

Discrete Mathematics, Richard Johnsonbaugh, 8th Edition, Prentice Hall, 2017.

Course Code: CSE2009	**Course Title: Computer Organization and Architecture**	**L-T- P- C**	3-0-0-3
Version No.	2.0		
Course Pre-requisites	CSE 2015 Digital Design		
Anti-requisites	**NIL**		
Course Description	This course introduces the core principles of computer architecture and organization from basic to intermediate level. This theory based course emphasizes on understanding the interaction between computer hardware and software. It equips the students with the intuition behind assembly-level instruction set architectures. It helps the students to interpret the operational concepts of computer technology as well as performance enhancement.		

Course Objective	The objective of the course is to familiarize the learners with the concepts of **Computer Organization and Architecture** and attain **Skill Development** through **Participative Learning** techniques.			
Course Outcomes	On successful completion of the course the students shall be able to: 1] Describe the basic components of a computer, their interconnections, and instruction set architecture [Comprehension] 2] Apply appropriate techniques to carry out selected arithmetic operations 3] Explain the organization of memory and processor sub-system			
Course Content:				
Module 1	Basic Structure of computers	Assignment	Data Analysis task	**12 Classes**
Topics: Computer Types, Functional Units, Basic Operational concepts, Bus Structures, Computer systems RISC & CISC, Performance – Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement. Arithmetic Operations on Signed numbers. Instructions and Instruction Sequencing, Instruction formats, Memory Instructions.				
Module 2	Instruction Set Architecture and Memory Unit	Assignment	Analysis, Data Collection	**12 Classes**
Topics: **Instruction Set Architecture:** Addressing Modes, Stacks and Subroutines. **Memory System:** Memory Location and Addresses, Memory Operations, Semiconductor RAM Memories, Internal Organization of Memory chips, Cache memory mapping Techniques.				
Module 3	Arithmetic and Input/output Design	Case Study	Data analysis task	**10 Classes**
Topics: **Arithmetic:** Carry lookahead Adder, Signed-Operand Multiplication, Integer Division, and Floating point operations. Input/output Design: Accessing I/O Devices, I/O communication, Interrupt Hardware, Direct Memory Access, Buses, Interface Circuits				
Module 4	BPU and Pipelining	Assignment	Analysis, Data Collection	**11 Classes**
Topics: **Basic Processing Unit:** Fundamental Concepts, Single Bus organization, Control sequence, Execution of a Complete Instruction, Multiple Bus Organization. **Pipelining:** Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, Hazards.				
Targeted Application & Tools that can be used: Targeted employment sector is processor manufacturing and memory chip fabrication vendors like Intel, AMD, Motorola, NVidia, Samsung, Micron Technology, western Digital etc. Targeted job profiles include Memory circuit design and verification engineers, Physical system design engineer, System programmer, Fabrication engineer etc.				

Tools:

Virtual Lab, IIT KGP

Tejas – Java Based Architectural Simulator, IIT Delhi

Text Book

Carl Hamacher, Zvonko Vranesic, Safwat Zaky, "Computer Organization", Fifth Edition, McGraw-Hill Higher Education, 2016 reprint.

References

William Stallings, "Computer Organization & Architecture – Designing for Performance", 11th Edition, Pearson Education Inc., 2019

David A. Patterson & John L. Hennessy, "Computer Organization and Design MIPS Edition- The Hardware/Software Interface", 6th Edition, Morgan Kaufmann, Elsevier Publications, November 2020.

Web References:

NPTEL Course on "Computer architecture and organization" IIT Kharagpur By Prof. Indranil Sengupta, Prof. Kamalika Datta. https://nptel.ac.in/courses/106105163

NPTEL Course on "Computer Organization", IIT Madras By Prof. S. Raman. https://nptel.ac.in/courses/106106092

https://puniversity.informaticsglobal.com:2229/login.aspx

Topics relevant to "SKILL DEVELOPMENT": Generation of Computers, CISC and RISC processors, Bus Arbitration, Collaboration and Data collection for Term assignments and Case Studies for Skill Development through Participative Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE3155	**Course Title: Data Communications and Computer Networks** **Type of Course: Program Core Theory–Laboratory integrated**	**L-T-P-C 3-0-2-4**	3	0	2	4
Version No.	1.0					
Course Pre-requisites	Digital Design					
Anti-requisites	NIL					
Course Description	The objective of this course is to provide knowledge in data communications and computer networks, its organization and its implementation, and gain practical experience in the installation, monitoring, and troubleshooting of LAN systems. . The associated laboratory is designed to implement and simulate various networks using Cisco packet tracer, NS2. All the lab exercises will focus on the fundamentals of creating multiple networks, topologies and analyzing the network traffics.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Data Communications and Computer Networks and attain **Employability** through **Problem Solving** Methodologies.					

Course Out Comes	On successful completion of the course, the students shall be able to: 1] I llustrate the Basic Concepts Of Data Communication and Computer Networks. 2] Analyze the functionalities of the Data Link Layer. 3] Apply the Knowledge of IP Addressing and Routing Mechanisms in Computer Networks. 4] Demonstrate the working principles of the Transport layer and Application Layer.			
Course Content:				
Module 1	Introduction and Physical Layer-CO1	Assignment	Problem Solving	07 Classes
Introduction to Computer Networks and Data communications, Network Components – Topologies, Transmission Media –Reference Models -OSI Model – TCP/IP Suite. Physical Layer -Analog and Digital Signals – Digital and Analog Signals – Transmission - Multiplexing and Spread Spectrum.				
Module 2	Reference Models and Data Link Layer – CO2	Assignment	Problem Solving	**7 Classes**
Data Link Layer - Error Detection and Correction – Parity, LRC, CRC, Hamming Code, Flow Control and Error Control, Stop and Wait, ARQ, Sliding Window, Multiple Access Protocols, CSMA/CD,CSMA/CA, IEEE 802.3, IEEE 802.11 Ethernet.				
Module 3	Network Layer – CO 3	Assignment	Problem Solving	**10 Classes**
Network Layer Services - Network Layer Services, Switching Techniques, IP Addressing methods- IPv4 IPV6 – Subnetting. Routing, - Distance Vector Routing – RIP-BGP-Link State Routing –OSPF-Multi cast Routing-MOSPF- DVMRP – Broad Cast Routing. EVPN-VXLAN, VPLS, ELAN.				
Module 4	Transport and Application Layer -CO3	Assignment	Problem Solving	**10 Classes**
Transport Layers - Connection management – Flow control – Retransmission, UDP, TCP, congestion control, – Congestion avoidance (DECbit, RED) The Application Layer: Domain Name System (DNS), Domain Name Space, SSH, FTP, Electronic Mail (SMTP, POP3, IMAP, MIME) – HTTP – – SNMP, Web Services, Virtual Networking.				

List of Laboratory Tasks:

Lab sheet -1, M-1, 3 [2 Hours]
Experiment No 1:
Level 1: Study of basic network commands and network configuration commands.

Lab sheet -2, M-1[2 Hours]
Experiment No 1:
Level 1: Identify and explore Network devices, models and cables. Introduction to Cisco packet tracer.
Experiment No. 2:
Level 2 – Create various network topologies using a cisco packet tracer.

Lab sheet -3, M-2,3 [2 Hours]
Experiment No. 1:
Level 2 - Basic Configuration of switch/router using Cisco packet tracer.
Experiment No. 2:
Level 2 -Configure the privilege level password and user authentication in the switch/router.

Lab sheet – 4, M-3 [2 Hours]
Experiment No. 1:
Level 2 - Configure the DHCP server and wireless router and check the connectivity

Lab sheet – 5, M-3 [2 Hours]
Experiment No. 1:
Level 2 - Configure the static routing in the Cisco packet tracer.
Experiment No. 2:
Level 2 - Configure the dynamic routing protocol in the Cisco packet tracer.

Lab sheet – 6, M-4 [2 Hours]
Experiment No. 1: Configuration of DNS Server with Recursive & Integrative approach in Cisco packet tracer.

Lab sheet – 7, M-4 [2 Hours]
Experiment No. 1:
Configure the telnet protocol in the router using the Cisco packet tracer.
Lab sheet – 8, M-4[2 Hours]
Experiment No. 1:
Level1- Introduction to NS2 and basic TCL program.
Lab sheet – 9, M-4 [2 Hours]
Experiment No. 1:
Level 1: Simulate three node Point to point network using UDP in NS2.
Experiment No. 2:
Simulate transmission of Ping message using NS2.
Lab sheet – 10, M-4[2 Hours]
Experiment No. 1:
Simulate Ethernet LAN using N-node in NS2.
Experiment No. 2:
Simulate Ethernet LAN using N-node using multiple traffic in NS2

Lab sheet –11, M-3,4 [2 Hours]
Experiment No. 1:
Level 1- Introduction to Wire Shark.
Experiment No. 2:
Level 2- Demonstration of packet analysis using wire shark.

Lab sheet –12, M-1,2,3 [2 Hours]
Experiment No. 1:
Level 2- Demonstration of switch and router configuration using real devices

Targeted Application & Tools that can be used: Cisco Packet Tracer, Wireshark, and NS2.

Case Study/Assignment: Choose and analyze a network from any organization/Assignment proposed for this course in CO1-CO4

Problem Solving: Choose and appropriate devices and implement various network concepts.
Programming: Simulation of any network using NS2.

Text Book

Behrouz A. Forouzan, "Data Communications and Networking 5E", 5th Edition, Tata McGraw-Hill, 2017.
Andrew S Tanenbaum, Nick Feamster & David J Wetherall, "Computer Networks" Sixth Edition, Pearson Publication, 2022

References

"Computer Networking: A Top-Down Approach", Eighth Edition, James F. Kurose, Keith W. Ross, Pearson publication, 2021.
William Stallings, Data and Computer Communication, 8th Edition, Pearson Education, 2007.
Larry L. Peterson and Bruce S. Davie: Computer Networks – A Systems Approach, 4th Edition, Elsevier, 2007.
E-Resources:
1.https://archive.nptel.ac.in/courses/106/105/106105183/
2. http://www.nptelvideos.com/course.php?id=393
3.https://www.youtube.com/watch?v=3DZLItfbqtQ
4.https://www.youtube.com/watch?v= fIdQ4yfsfM
5. https://www.digimat.in/keyword/106.html
https://puniversity.informaticsglobal.com/login

Course Code: CSE3146	**Course Title: Advanced Java Programming** **Type of Course:1] School Core** **2] Laboratory integrated**	**L- T-P- C**	1	**0**	**4**	**3**
Version No.	1.0					

Course Pre-requisites	**[1] Problem Solving Using Java (CSE1001) [2] Database ManagementSystem (CSE2074) [3] Web Technology (CSE2006)** **Basic Knowledge about DBMS, Knowledge on Core Java (OOPs Principles), Client-serverArchitecture, HTML**			
Anti-requisites	**NIL**			
Course Description	**The purpose of this course is to introduce the students to Java Advanced API enhanced byDesign Patterns and SOLID Principles. The course is both conceptual and analytical and is understood with JDK 8 software & IntelliJ IDE. This course develops critical thinking skills by augmenting the student's ability to develop distributed model for control of various modern management systems like banking management system, student information management system, , Library Management System etc. w*ith* the necessary API for communication with database enhanced by the current industrial approach of Java's SOLID principle and design patterns. This course also involves essential core java concepts like multithreading, file handling, event handling etc.**			
Course Objectives	This course is designed to improve the learners' EMPLOYABILITY SKILLS by usingEXPERIENTIAL LEARNING techniques. Please add as per what the course covers in the criteria1 NAAC Template.			
Course Outcomes	On successful completion of this course the students shall be able to: 1. Explain the benefits of Design-Pattern & SOLID principle in java based applications. 2. Understand Concurrent Programming using Java Multi-Threading. 3. Apply Communication mechanisms of Java with DBMS. 4. Implement Web MVC application using Servlet and JSP Technology. 5. Test JPA Implementation using Hibernate.			
Course Content:				
Module 1	Multi- Threading (Comprehension)	Assignment	Knowledge Ability	**11 Hours**
Topics: **Multi-Threading in Java:** Understanding Threads , Needs of Multi-Threaded Programming ,Thread Life-Cycle, Thread Priorities ,Synchronizing Threads, Inter Communication of Threads ,Critical Factor in Thread –DeadLock, The Executor Framework.				
Module 2	Input & Output Operation inJava (Comprehension)	Assignment	File Operations	**11 Hours**
Topics: **Java I/O Operations :** Input/Output Operation in Java(java.io Package),Streams and the new I/O Capabilities ,Understanding Streams, Working with File Object, File I/O Basics, Reading and Writing to Files, Buffer and Buffer Management, Read/Write Operations with File Channel, Serializing Objects, Observer and Observable Interfaces.				

Module 3	Collection and Database programming using JDBC (Comprehension)	Assignment	Data Storage	**12 Hours**
Topics: **Collection** - The Collection Framework : Collections of Objects , Collection Types, Sets , Sequence, Map, Understanding Hashing, Uses of ArrayList & Vector , Comparable and Comparator Interfaces. **Database Programming using JDBC**- Introduction to JDBC, JDBC Drivers & Architecture, CRUD operation UsingJDBC, Connecting to non-conventional Databases.				
Module 4	Distributed Programming withServlet (Application)	Assignment	Distributed Programming	**11 Hours**
Topics: **Servlet** - Web Application Basics, Architecture and challenges of Web Application, Introduction to servlet, Servlet life cycle, Developing and Deploying Servlets, Create and compile servlet source code, start tomcat, start a web browser and request the servlet, servlet API, Handling HTTP Requests and Responses: Handling HTTP GET requests and POST request, Session Tracking, Simple Servlet Program to fetch database records				
Module 5	Distributed Programming withJSP (Application), Introduction to Spring Framework (Application)	Assignment	Distributed Programmin g	**11 Hours**

Topics:

JSP - Introduction to JSP, Creating simple JSP Programs, How JSP is processed, JSP Scripting Constructs, Predefined Variables, JSP Directives, Simple JSP Program to fetch database records.
Spring CORE, Overview of Spring, Spring Architecture, bean life cycle, Java and XML Configuration on Spring, SpringDifferent Modules.
Spring JPA, JPA Specification, Classes and Interfaces, Object Relational Mapping using JPA, JPA implementation with Hibernate, Simple JPA-Hibernate program to Create Database schemas.

List of Laboratory Tasks:

Labsheet -1 [4 + 1 Practical Sessions]
Experiment No 1:
Level 1: Demonstration of Thread Class and Runnable Interface.
Level 2 – Implementation of Producer-Consumer Problem.
Labsheet -2 [3 +1 Practical Sessions]
Experiment No. 1:
Level 1 – Usages of Java.io.* package.
Level 2 – File operations with a case study.

Labsheet – 3 [3 +1 Practical Sessions]
Experiment No. 1:
Level 1 – Practicing classes and methods in java.util.collection.
Level 2 – Scenario based questions to apply all collections. [Group wise]

Labsheet – 4 [3 + 1 Practical Sessions]
Experiment No. 1:
Level 1 – JDBC complete Demonstration with Student Database
Level 2 – Implementation of Student Information Management (Standalone). [Group wise]

Labsheet – 5 [3 + 1 Practical Sessions]
Experiment No. 1:
Level 1 – Web page creation using HTML, Dynamic web page using java.servlet and JDBC
Level 2 – Implementation of Student Information Management (WEB based). [Group wise]

Labsheet – 6 [3 + 1 Practical Sessions]
Experiment No. 1:
Level 1 – Web page creation using HTML, Dynamic web page using java.servlet , JSP and JDBC
Level 2 – Implementation of Student Database using JPA Hibernate

Targeted Application & Tools that can be used: Java 8 / MYSQL 8 / Eclipse /IntelliJ (IDE)

Project work/Assignment: Mention the Type of Project /Assignment proposed for this course

Build a Standalone database application using Java Swing as Front End. Indicative areas include; TimeTable Management, Student Expense Tracker, Important Mail Fetcher, etc.
Build a real time database application using J2EE as Front End. Indicative areas include; health care, education,industry, Library, Transport and supply chain, etc.

Text Books

1. Cay S Horstmann and Gary Cornell, “CORE JAVA volume II-Advanced Features, 9th Edition.

References

1. Herbert Schildt, *"Java 2: The Complete Reference"*, Tata McGraw-Hill Education,6th Edition.
2. Y.Daniel Liang, "Introduction to Java programming Comprehensive Version", Pearson Education, 10thEdition.
3. Core and Advanced Java Black Book, Dream Tech Press.
4. Spring in Action , Graig Walls, 5th Edition
5. Java Persistence with Hibernate , Christian Bauer & Gavin King, 2nd Edition
6. https://www.youtube.com/watch?v=JGNTYXkVCVY&list=PLd3UqWTnYXOkTSBCBNyyhxo_jxlY_uTWA&ind ex=2

urse Code: S4004	**urse Title: Aptitude Training- Intermediate be of Course: Practical Only Course**		**P- C**	0	0	2	1
rsion No.						
urse Pre-requisites	Students should have the basic concepts of Quantitative aptitude along with its applications in real life problems.					
ti-requisites						
urse Description	This is a skill-based training program for the students (Undergraduate). This course is designed to enable the students to enhance their skills in Quantitative Aptitude.					
urse Objective	The objective of the course is to familiarize the learners with the concepts of Aptitude and attain Skill Development through Problem Solving techniques.					
urse Outcomes	On successful completion of the course the students shall be able to: CO1] **Understand** all the concepts. CO2] **Apply** the concepts in problem solving (Bloom's taxonomy Level 3)					
urse Content:						
dule 1	antitative Ability	ignment				**24 Hours**
pics: mber System, Percentage, Ratio and Proportion, Average, Mixture and Allegation, Time and Work, Profit and Loss, Time Speed and Distance, Boats and Streams, Simple Interest and Compound Interest, Probability, Permutation and Combination.						
geted Areas plication area: Placement activities and Competitive examinations. ols: LMS						
t Book t Track Objective by Rajesh Verma Aggarwal kesh Yadav						

ferences ww.indiabix.com ww.testbook.com ww.youtube.com/c/TheAptitudeGuy/videos
Topics relevant to Skill development: Quantitative aptitude for Skill Development through Problem solving Techniques. This is attained through assessment component mentioned in course handout.
Evaluation – Continuous Evaluation (Topic wise evaluation Mid-Term & End term)

Course Code: CSE3216	**Course Title:** Mastering Object- Oriented Concepts in Python **Type of Course:** Lab		**L- T- P- C**	0-0-2-1
Version No.	1			
Course Pre-requisites	CSE1005 – Programming in Python			
Anti-requisites	NIL			
Course Description	This course covers mastering object-oriented concepts in Python, including classes, inheritance, polymorphism, and encapsulation. Students will learn to design and implement robust, reusable code using real-world examples. Ideal for those with basic Python knowledge, it enhances problem-solving skills and software development proficiency.			
Course Objective	**The objective of the course is to familiarize the learners with the concepts of Mastering Object Oriented Concepts in Python and attain Skill Development through Experiential Learning.**			
Course Out Comes	**CO1:** Explain features of Oops along with creation of Python classes and objects to represent real world Objects. **[Understand]** **CO2:** Demonstrate inheritance, polymorphism, and abstraction in Python to build maintainable and extendable software systems.**[Apply]** **CO3:** Demonstrate exception handling in Python to build robust error-handling mechanisms and debugging tool and Assess various file handling techniques in Python. **[Apply]**			
Course Content:				
Module 1	**Introduction to OOPS, Classes and Objects**	MCQ	Assignment	**10 Sessions**
Topics: **Introduction to OOPs:** Problems in Procedure Oriented Approach, Specialty of Python Language, Features of OOPS - Classes and Objects, Encapsulation, Abstraction, Inheritance and Polymorphism. **Classes and Objects:** Creating a Class, The Self Variable, Constructor, Destructors, Types of Variables, Namespaces, Types of Methods - Instance Methods, Class Methods, Static Methods,				

Passing Members of One Class to Another Class, Inner Classes.				
Module 2	**Inheritance and Polymorphism**	MCQ	Assignment	**10 Sessions**
Constructors in Inheritance, Overriding Super Class Constructors and Methods, The Super() Method, Types of Inheritance – Single Inheritance, Multiple Inheritance, Method Resolution Order(MRO), Polymorphism, Duck Typing Philosophy of Python, Operator Overloading, Method Overloading, Method Overriding. **Abstract Classes and Interfaces:** Abstract Method and Abstract Class, Interfaces in Python, Abstract Classes vs. Interfaces.				
Module 3	**Exceptions and Files in Python**	MCQ	Assignment	**10 Sessions**
Exceptions: Errors in a Python Program – Compile-Time Errors, Runtime Errors, Logical Errors. Exceptions, Exception Handling, Types of Exceptions, The Except Block, The assert Statement, User-Defined Exceptions, Logging the Exceptions. **Files in Python:** Files, Types of Files in Python, Opening a File, Closing a File, Working with Text Files Containing Strings, Knowing whether a File Exists or Not, Working with Binary Files, The with Statement, Pickle in Python, The seek() and tell() Methods.				
Targeted Application & Tools that can be used: Python, PyCharm				
Project work/Assignment:				
Assignment: Module 1 Assignment: Design and implement a Python application that simulates a banking system using classes and methods for customers and accounts. Module 2 Assignment: Develop a Python application that simulates Library management system that demonstrates inheritance, polymorphism and abstraction concepts. Module 3 Assignment: Develop a Python program that handles different types of exceptions while processing user input for a movie ticket booking system showcasing exception handling and File handling concepts.				
Text Book Dr. R Nageshwara Rao, “Core Python Programming”, Dreamtech Press, 3rd Edition, 2021.				
References Alex Martelli, Anna Ravenscroft & Steve Holden, “Python in a Nutshell The Definitive Reference”, O'Reilly Media, 3rd edition, 2017. Luciano Ramalho, “Fluent Python Clear, Concise, and Effective Programming”, O'Reilly Media, 2nd edition, 2022. Mark Lutz, “Learning Python: Powerful Object-Oriented Programming”, O'Reilly Media, 5th edition, 2013. David Beazley, Brian K. Jones, “Python Cookbook: Recipes for Mastering Python 3”, O'Reilly Media, 3rd edition, 2013. **Weblinks:** www.learnpython.org https://realpython.com/python3-object-oriented https://www.tutorialspoint.com/python/python_oops_concepts.htm				
Topics relevant to “SKILL DEVELOPMENT”: Building Real-World Applications Using OOPS Concepts, Error Handling and Debugging Techniques, Concurrency in Python, Advanced File Handling Techniques, Creating and Managing Python Packages and Modules, Designing and Implementing Python Interfaces **This is attained through assessment component mentioned in course handout.**				

Course Code: **CSE1700**	**Course Title:** Essentials of AI **Type of Course:** Theory and Lab	**L- T-P-C**	3	**0**	0	**3**
Version No.	2.0					
Course Pre-requisi Data **tes**	Basic knowledge of programming, mathematics, understanding of data handling					
Anti-requisites	**NIL**					
Course Description	This course is a comprehensive introductory course designed to equip learners with the fundamental Python programming skills necessary to work with artificial intelligence (AI) technologies. This course is aimed at individuals who are new to AI but have a basic understanding of programming concepts. It combines Python programming fundamentals with hands-on experience in implementing AI techniques such as machine learning, neural networks, and natural language processing.					
Course Objective	The objective of the course is to Understand Python Programming Fundamentals, Manipulate and Process Data with Python, Implement Machine Learning Algorithms and Build and Train Neural Networks for AI Applications.					
Course Outcomes	On successful completion of the course the students shall be able to: CO 1: Apply Python Programming to AI Projects CO 2: Build and Train Machine Learning Models CO 3: Develop Deep Learning Models with Neural Networks CO 4: Deploy AI Solutions and Understand Ethical Implications					
Course Content:						
Module 1	Introduction to Python Programming for AI	Assignment	Implementation	**10 Sessions**		
Topics: Python Basics: Variables, Data Types, Operators, and Control Flow Functions, Loops, and Conditionals statements, Data Structures: Lists, Tuples, Dictionaries, Sets ,Introduction to Libraries: NumPy and Pandas for data manipulation, Basic Input/Output and File Handling Introduction to Python for AI: Libraries and Frameworks Overview						
Module 2	Data Processing, Visualization	Assignment	Implementation	10 Sessions		
Topics:						

cleaning and preprocessing with Pandas,Handling missing data, outliers, and duplicates, Data transformation (Normalization, Encoding), Introduction to Matplotlib and Seaborn for Data Visualization, Exploratory Data Analysis (EDA), Visualizing datasets to understand patterns and relationships.

Module 3	Introduction to Machine Learning	Mini - Project	Implementation	10 Sessions

Topics:

What is Machine Learning? Types of ML algorithms Supervised Learning: Regression, Classification, Unsupervised Learning: Clustering, Key ML Algorithms: Linear Regression, Decision Trees, K-Means ,Introduction to Scikit-learn library

Model evaluation (Accuracy, Precision, Recall, Confusion Matrix)

Module 4	Neural Networks and Deep Learning	Quiz	Implementation	10 Sessions

Topics:

Introduction to Neural Networks and Deep Learning, Perceptron Model and Backpropagation

Deep Neural Networks and Activation Functions, Introduction to TensorFlow and Keras, Building and Training Neural Networks for Image and Text Classification, Overview of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs)

Targeted Application & Tools that can be used:

Applications:

Data Preprocessing: Clean and manipulate data from various sources such as CSV, Excel, SQL databases, and APIs.

Exploratory Data Analysis (EDA): Gain insights into datasets by identifying trends, patterns, and outliers.

Predictive Modeling: Build models for classification (e.g., spam detection) and regression (e.g., house price prediction).

Clustering: Group data into clusters for unsupervised learning tasks (e.g., customer segmentation).

Model Evaluation: Assess model performance using appropriate metrics such as accuracy, precision, recall, and F1-score.

Tools:

Pandas: For data manipulation and cleaning (e.g., handling missing values, merging datasets).

NumPy: For numerical operations and working with arrays and matrices.

Matplotlib: For creating static, animated, and interactive visualizations.

Seaborn: For advanced data visualizations (e.g., heatmaps, pair plots).

Plotly: For creating interactive visualizations, especially useful for large datasets.

Scikit-learn: The go-to library for implementing machine learning algorithms (e.g., linear regression, decision trees, k-means clustering).

XGBoost: For advanced gradient boosting models, particularly for large-scale machine learning tasks.

TensorFlow (for deep learning in Module 4): A powerful open-source library for building machine learning and deep learning models.

Keras: High-level neural network API, built on top of TensorFlow, to easily create deep learning models.

NLTK: The Natural Language Toolkit for various text processing tasks like tokenization, stemming, and part-of-speech tagging.

spaCy: A fast NLP library for advanced NLP tasks such as named entity recognition and dependency parsing.

Transformers (by Hugging Face): A powerful library for using pre-trained Transformer-based models like BERT, GPT, and others for advanced NLP tasks.

Text Book(s):

T1: **Essentials of Python for Artificial Intelligence and Machine Learning** by Pramod Gupta and Anupam Bagchi

Reference(s):

"Artificial Intelligence with Python" – Prateek Joshi

"Python Machine Learning" – Sebastian Raschka & Vahid Mirjalili

"Hands-On Artificial Intelligence with Python" – Teet Straus

"Deep Learning for Coders with Fastai and PyTorch" – Jeremy Howard & Sylvain Gugger

Course Code: **CSE2508**	**Course Title:** Mobile Applications and Development **Type of Course:** Theory	**L- T-P- C**	2	**0**	0	**2**
Version No.	2.0					
Course Pre-requisites	CSE3514 Object Oriented Programming Using Java					
Anti-requisites	**NIL**					
Course Description	The course deals with the basics of android platform and application life cycle. The goal of the course is to develop mobile applications with Android containing at least one of the following phone material components: GPS, accelerometer or phone camera, use simple GUI applications and work with database to store data locally or in a server. Topics include user interface design; user interface building; input methods; data handling; network techniques and URL loading; GPS and motion sensing. Android application framework and deployment. Power management, Screen resolution, Touch interface, Store data on the device.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Mobile Applications and Development as mentioned above and attain Employability Skills through Experiential Learning Techniques.					
Course Outcomes	On successful completion of the course the students shall be able to: 1. Discuss the fundamentals of mobile application development and its architecture. (Comprehension) 2. Illustrate mobile applications with appropriate android view. (Application) 3. Demonstrate the use of services, broadcast receiver, Notifications and content provider.(Application) 4. Apply data persistence techniques, to perform CRUD operations. (Application) 5. Use advanced concepts for mobile application development. (Application)					
Course Content:						
Module 1	Introduction and Architecture of Android	Assignment	Simulation/Data Analysis	5 Sessions		
Topics: Android: History and features, Architecture, Development Tools, Android Debug Bridge (ADB), and Life cycle.						
Module 2	User Interfaces, Intent and Fragments	Term paper/Assignment	Simulation/Data Analysis	6 Sessions		

Topics: Views, Layout, Menu, Intent and Fragments.				
Module 3	Components of Android	Term paper/Assignment	Simulation/Data Analysis	6 Sessions
Topics: Activities, Services, Broadcast receivers, Content providers, User Navigation				
Module 4	Notifications and Data Persistence	Term paper/Assignment	Simulation/Data Analysis	6 Sessions
Topics: Notification, Shared Preferences, SQLite database, Android Room with a View, Firebase.				
Module 5	Advance App Development	Term paper/Assignment	Simulation/Data Analysis	7 Sessions
Topics: Graphics and Animation, App Widgets, Sensors, Performance, Location, Places, Mapping, Custom Views, Canvas.				

Targeted Application & Tools that can be used:

Applications:

Native Android Applications

Native iOS Applications

Cross Platform mobile Apps

Mobile web Applications

Text Book(s):

T1. Pradeep kothari “Android Application Development - Black Book”, dreamtechpress

T2. Barry Burd (Author), “Android Application Development” ALL – IN – ONE FOR Dummies

T3. Jeff Mcherter (Author),Scott Gowell (Author), “Professional mobile Application

Development” paperback, Wrox - Wiley India Private Limited

T4. Wei-Meng Lee (Author) “Beginning Android Application Development” Wrox – Wiley

India Private Limited

Reference(s):

1. Bill Phillips, Chris Stewart, and Kristin Marsicano (Author) “Android Programming” 3rd edition, 2017.The Big Nerd Ranch Guide, Big Nerd Ranch LLC, 5. The Big Nerd Ranch Guide, by”

2. Erik Hellman, “Android Programming – Pushing the Limits”, 1st Edition, Wiley India Pvt Ltd, 2014.

3. Dawn Griffiths and David Griffiths, “Head First Android Development”, 1st Edition, O’Reilly SPD Publishers, 2015.

4. J F DiMarzio, “Beginning Android Programming with Android Studio”, 4th Edition, Wiley India Pvt Ltd, 2016. ISBN-13: 978-8126565580

5. Anubhav Pradhan, Anil V Deshpande, “ Composing Mobile Apps” using Android, Wiley 2014, ISBN: 978-81-265-4660-2

6. Reto Meier “Professional Android Application Development”

E-Resources: https://puniversity.informaticsglobal.com/login Or http://182.72.188.193/

Course Code: CSE2500	**Course Title: Theory of Computation** **Type of Course: Theory Only**	**L- T-P- C**	3	0	0	3
Version No.	2.0					
Course Pre-requisites	The students should have the Knowledge on Set Theory					
Anti-requisites	Nil					
Course Description	The course deals with introduction of formal languages and the correspondence between language classes and the automata that recognize them. Topics include: Formal definitions of grammars and acceptors, Deterministic and Nondeterministic systems, Grammar ambiguity, finite state and push-down automata; normal forms; Turing machines and its relations with algorithms.					

Course Objective	The objective of the course is to familiarize the learners with the concepts of **Theory of Computation as mentioned above** and attain **Skill Development** through **Problem Solving** Methodologies.			
Course Out Comes	On successful completion of the course the students shall be able to: 1. Describe various components of Automata. (Knowledge) 2. Illustrate Finite Automata for the given Language. (Application) 3. Distinguish between Regular grammar and Context free grammar. (Comprehension) 4. Construct Push down Automata. (Application) 5. Construct Turing machine for a Language. (Application)			
Course Content:				
Module 1	**Introduction to automata theory**	Assignment	Problems on Strings and Language operations	**06 Sessions**
Topics: Introduction to Automata Theory, Applications of Automata Theory, Alphabets, Strings, Languages & operations on languages, Representation of automata, Language recognizers, Finite State Machines (FSM): Deterministic FSM, Regular languages, Designing FSM, Nondeterministic FSMs				
Module 2	**Finite Automata**	Assignment	Problems on DFA, NFA's	**13 Sessions**
Topics: Basic concepts of Finite automata, DFA- definitions of DFA, Deterministic Accepters Transition Graphs and Languages and DFA's, Regular Languages, NFA- Definition of a Nondeterministic Accepter, Languages and NFA's Why Non-determinism? Equivalence of Deterministic and Nondeterministic Finite Accepters, Reduction of the Number of States in Finite Automata.				
Module 3	**Regular Expressions & Context Free Grammar**	Assignment	Problems on RE, CFG, PT, PL and Ambiguity	**12 Sessions**
Topics: Formal Definition of a Regular Expression, Languages Associated with Regular Expressions, Languages, Regular Languages (RL) and Non-regular Languages: Closure properties of RLs, to show some languages are not RLs, Closure Properties of Regular Context Free Grammars-Examples of Context-Free Languages, Leftmost and Rightmost Derivations, Derivation Trees, Relation Between Sentential Forms and Derivation Trees, Ambiguity in Grammars and Languages: Ambiguous Grammars, Removing Ambiguity, Chomsky Normal Form, Gribiche Normal Form.				
Module 4	**Push down Automata**	Assignment	Problems on pushdown Automaton	**08 Sessions**
Topics: Definition of a Pushdown Automaton, Language Accepted by a Pushdown Automaton, Acceptance by Final State, Acceptance by Empty Stack, From Empty Stack to Final State, From Final State to Empty Stack Equivalence of PDA's and CFG's: From Grammars to Pushdown Automata.				
Module 5	**Turing Machine**	Assignment	Problems on Turning Machine	**07 Sessions**
Topics: Definition of a Turing Machine, Turing Machines as Language Accepters, Example Languages to construct Turing machine, Turing Machines as Transducers, Halting Programming Techniques for Turing Machines				
Targeted Application & Tools that can be used: Targeted Application: 1. Text Processing 2. Compilers 3. Text Editors 4. Robotics Applications				

5. Artificial Intelligence

Tools:

1. JFLAP (Java Formal Language and Automata Package) Software simulation tool. It's interactive educational software written in Java to experiment topics in automata theory.
2. Turing machine Online simulators.

Text Book

1. Peter Linz, "An introduction to Formal Languages and Automata", Jones and Bartlett Publications 6th Ed, 2018.

References

1. Aho, Ullman and Hopcroft, "Theory of Computation", Pearson India 3rd Edition 2008.
2. Michael Sipser, "Theory of Computation", Cengage India 3rd Ed, 2014.

E-Resources

NPTEL course – https://onlinecourses.nptel.ac.in/noc21_cs83/preview

Topics relevant to "SKILL DEVELOPMENT": Deterministic and Non-Deterministic Automaton, Regular Expressions, CFGs, Turning Machine and Pushdown automaton for Skill Development through Problem Solving methodologies. This is attained through assessment component mentioned in course handout.

urse Code: E2513	**urse Title: Object Oriented Analysis and Design** **pe of Course: Program Core and Theory**	**- T-P-C**				
rsion No.						
urse Pre-requisites	E 1006 Problem Solving using Java					
ti-requisites	**L**					
urse Description	This course covers the analysis and design methodology in sufficient depth to convey a good understanding of object-oriented analysis and design using the unified process. Students will be able to design a use case model, identify the classes and their responsibilities, use interaction models to capture the interdependence among objects/classes and design an efficient solution. The application of the design axioms and the iterative nature of the process are emphasized. This course will enable students to apply object oriented concepts in all the stages of the software development life cycle.					
urse objective	This course is designed to improve the learners " SKILL DEVELOPMENT "by using EXPERIENTIAL LEARNING techniques.					
urse tcomes	On successful completion of the course the students shall be able to: 1]Describe the basics of object oriented system development **[Knowledge]** 2]Identify the various techniques for object-oriented analysis techniques [**Comprehension]** 3] Apply the design axioms to create appropriate UML diagrams. [**Application]** 4]Apply the design process to develop implementation models. **[Application]**					
urse Content:						
dule 1	roduction to Object oriented system	signment	ntify problem domain objects for an application domain	**9 Classes**		
pics: Object Basics-Object Oriented System Development Life Cycle- Use case driven approach-Rumbaugh Object Model- Booch Methodology-Jacobson Methodology-Unified Approach, Static and Dynamic Modeling-Unified Modeling Language						
dule 2	ject oriented analysis	signment	ntification of candidate classes using various approaches	**12 Classes**		
pics: Identifying use cases-Object Analysis-Classification: Theory-Approaches for Identifying Classes: Noun Phrase approach, Common Class pattern approach, Use case driven approach, Classes, Responsibilities and Collaborators- Identifying Object relationships: Associations, Super–sub class relationships, Aggregation. UML diagrams: Use case Diagram , Class diagram.						
dule 3	roduction to	signment	ply axioms to create class	**12 Classes**		

	axiomatic design		diagram	

pics:

Object Oriented Design Axioms-Designing Classes -Class visibility -Redefining attributes - Designing methods and protocols -Packages and managing classes, UML Diagrams: Interaction diagram, Sequence diagram, Collaboration diagram, State-chart diagram, Activity diagram

dule 4	ject oriented Design process	signment	ply the design process and develop a component and deployment diagram.	**12 Classes**

pics:

Access Layer- Object Storage Persistence - Object oriented Database System-Designing view layer classes -Macro level process -Micro level process- Prototyping the user interface UML diagrams: component diagram, Deployment diagram, Quality Assurance Tests-Testing Strategies.

ols that can be used:

ols:

crosoft visio, Rational software architect(RSA)

goUML, Rational Rose, StarUML, Umbrello

ject work/Assignment:

rm Assignments:

ntify Use Cases and develop the Use Case model

ntity the conceptual classes and develop a UML Class diagram

ing the identified scenarios find the interaction between objects and represent them using UML Interaction diagrams

ntify the business activities and develop an UML Activity diagram

xt Book

1. Ali Behrami, "Object Oriented Systems Development using Unified Modeling Language" McGraw Hill International Edition, July 2017.

ferences

1.Craig Larman, "Applying UML and Patterns", Pearson Education.
2.Grady Booch, "Object Oriented Analysis and Design with Applications", Addison-Wesly.
3.Simon Bennett, Steve McRobb, Ray Farmer, "Object Oriented Systems Analysis and Design using UML", McGrawHill Education

Topics related to development of "FOUNDATION": Object Basics-Object Oriented System Development Life Cycle- Use case driven approach-Rumbaugh Object Model- Booch Methodology-Jacobson Methodology-Unified Approach, Static and Dynamic Modeling-Unified Modeling Language

Topics related to development of " SKILL DEVELOPMENT": UML diagrams: Use case Diagram , Class diagram, Interaction diagram, Sequence diagram, Collaboration diagram, State-chart diagram, Activity diagram, component diagram, Deployment diagram using the tool StarUML software

Course Code: CSE2502	**Course Title: Operating Systems** **Type of Course: Program Core and Theory Only**		**L-T- P- C**	3	0	0	3
Version No.	1.0						
Course Pre-requisites	CSE2009- Computer Organization, Problem solving using C Students should have basic knowledge on computers, computer software & hardware, and Computer Organization. Prior programming experience in C is recommended.						
Anti-requisites	**NIL**						
Course Description	This course introduces the concepts of operating system operations, operating system structure and its design and implementation. It covers the classical operating systems internal algorithms such as process scheduling, synchronization, deadlocks detection and recovery and memory management. The course also enhances the problem solving, systems programming ability and case studies.						
Course Object	The objective of the course is to familiarize the learners with the concepts of Operating Systems and attain **Employability** through **Problem Solving** Methodologies.						
Course Out Comes	On successful completion of the course the students shall be able to: 1] Describe the fundamental concepts of operating Systems and case studies. [**Knowledge**] 2] Demonstrate various CPU scheduling algorithms. .[**Application**] 3] Apply various tools to handle synchronization problems.[**Application**] 4] Demonstrate deadlock detection and recovery methods **[Application]** 5] Illustrate various memory management techniques.[**Application**]						
Course Content:							
Module 1	Introduction to Operating System	Assignment	Programming				**9 Hours**
Topics: Introduction to OS , Operating-System Operations, Operating System Services, , System Calls and its types, Operating System Structure, System Program and its types, Linkers and Loaders, Overview of OS design and implementation, Open-source operating system							
Module 2	Process Management	Assignment/Case Study	Programming/Simulation				**11 Hours**
Topics: Process Concept, Operations on Processes, Inter Process Communication, Communication in client-server systems (sockets, RPC, Pipes), Introduction to threads - Multithreading Models, Thread Libraries, Threading Issues, Process Scheduling– Basic concepts, Scheduling Criteria, Scheduling Algorithms: FCFS, SJF, SRTF, RR and Priority.							
Module 3	Process Synchronization and Deadlocks	Assignment	Programming				**11 Hours**
Topics: The Critical-Section Problem- Peterson's Solution, Synchronization hardware, Semaphores, Classic Problems of Synchronization with Semaphore Solution- Producer-Consumer Problem, Reader-Writer							

problems, Dining Philosopher's Problem, . Introduction to Deadlocks, Necessary conditions for deadlock, Resource allocation Graph, Methods for handling deadlock: Deadlock Prevention and Implementation, Deadlock Avoidance and Implementation, Deadlock detection & Recovery from Deadlock.

Module 4	Memory Management	Assignment	Programming/Simulation	**10 Hours**

Topics:
Introduction to Memory Management, Basic hardware-Base and Limit Registers, Memory Management Unit(MMU), Dynamic loading and linking, Swapping, Contiguous and Non-Contiguous Memory Allocation, Segmentation, Paging - Structure of the Page Table – Virtual Memory and Demand Paging – Page Faults and Page Replacement Algorithms, Copy-on-write, Allocation of Frames, Thrashing
Introduction to File system management: File System Interface (access methods, directory structures), File system implementation.

Targeted Application:
Application area is traffic management system, banking system, health care and many more systems where in there are resources and entities that use and manage the resources.

Software Tools:

Oracle Virtual Box/VMWare Virtualization software [Virtual Machine Managers]. Used to install and work on multiple guest Operating Systems on top of a host OS.

Intel Processor identification utility: This software is used to explain about multi-core processors. It helps to identify the specifications of your Intel processor, like no of cores, Chipset information, technologies supported by the processor etc.

Project work/Assignment

Demonstrate process concepts in LINUX OS.
Simulation of CPU scheduling algorithms.
Develop program to demonstrate use of Semaphores in threads.
Develop program to demonstrate use of deadlock avoidance algorithms.
Develop program to demonstrate use of page replacement algorithms.
Simulation of memory allocation strategies [first fit, best fit and worst fit].

Text Book

Silberschatz A, Galvin P B and Gagne G , "Silberschatz's Operating System Concepts", Paperback, Global Edition Wiley, 2019

References

Silberschatz A, Galvin P B and Gagne G, "Operating System Concepts", 10th edition Wiley, 2018.
William Stallings, "Operating Systems",Ninth Edition, By Pearson Paperback ,1 March 2018.
Sundaram RMD, Shriram K V, Abhishek S N, B Chella Prabha, " Cracking the Operating System skills", Dreamtech, paperback, 2020
Remzi H. Arpaci-Dusseau Andrea C. Arpaci-dusseau , "Operating Systems: Three Easy Pieces, Amazon digital Services", September 2018.

E-resources/Weblinks

https://www.os-book.com/OS9/
https://pages.cs.wisc.edu/~remzi/OSTEP/
https://codex.cs.yale.edu/avi/os-book/OS10/index.html

Course Code: **CSE1701**	**Course Title:** Essentials of AI LAB **Type of Course:** Lab	**L- T-P-C**	0	**0**	4	**2**
Version No.	2.0					
Course Prerequisites	Basic Java Programming Knowledge, Mathematics: Linear Algebra and Probability, Basic Data Structures and Algorithms, Familiarity with Libraries and Tools, Understanding of Basic Machine Learning Concepts.					
Anti-requisites	**NIL**					
Course Description	This course introduces students to the essential concepts and techniques of Artificial Intelligence (AI) with a focus on practical implementation using Python. Students will explore core AI topics such as search algorithms, knowledge representation, machine learning, and neural networks, while gaining proficiency in using popular Python libraries like NumPy, pandas, scikit-learn, and TensorFlow. Through a series of lab exercises and projects, students will apply AI principles to solve real-world problems, develop intelligent applications, and understand how AI systems function at a foundational level.					
Course Objective	The primary objectives of the course are to Gain Proficiency in AI Concepts and Python Implementation, Develop and Implement Machine Learning Models, Understand and Build Neural Networks, Apply AI to Real-World Problems					
Course Outcomes	On successful completion of the course the students shall be able to: Proficiency in Implementing AI Algorithms Using Python Ability to Build and Evaluate Machine Learning Models Hands-on Experience with Neural Networks and Deep Learning Practical Application of AI to Solve Real-World Problems					
Course Content:						
Module 1	nIntroduction to AI and ython for AI	ssignment	mplementation		**10 Sessions**	
Lab Assignment 1: Setting Up the Python Environment **Objective:** Get familiar with setting up a Python environment for AI projects. **Tasks:**						

Install Python, Anaconda, and Jupyter Notebook.

Set up a virtual environment for AI development.

Install essential Python libraries: numpy, pandas, matplotlib, and scikit-learn.

Write and execute simple Python code to verify installation (e.g., print a “Hello AI” message).

Lab Assignment 2: Basic Python Programming for AI

Objective: Understand and practice the basic Python syntax and data structures used in AI.

Tasks:

Write Python code to work with basic data types (integer, float, string, boolean).

Implement and manipulate Python lists, tuples, sets, and dictionaries.

Create basic control flow structures: if-else, for loops, while loops.

Use functions and lambda functions to solve small AI-related problems, such as calculating factorial or Fibonacci numbers.

Lab Assignment 3: Data Exploration and Preprocessing

Objective: Learn how to work with data for AI models.

Tasks:

Load a dataset (e.g., Titanic or Iris dataset) using pandas.

Clean the dataset by handling missing values, removing duplicates, and converting data types if needed.

Explore the dataset by visualizing it using matplotlib and seaborn.

Perform basic data preprocessing tasks such as feature scaling, encoding categorical variables, and splitting data into training and testing sets.

Module 2	Data Processing, Visualization	Assignment	Implementation	10 Sessions

Lab Assignment 1: Data Preprocessing with Pandas

Objective:
Learn the fundamentals of data preprocessing, including cleaning, handling missing values, and performing basic transformations using **Pandas**.

Tasks:

Load and Inspect the Dataset:

Load a dataset (e.g., **Iris**, **Titanic**, **Wine Quality** dataset) using pandas.read_csv() or pandas.read_excel().

Inspect the first few rows of the dataset using .head() and check basic information using .info().

Handle Missing Values:

Identify missing values in the dataset using .isnull() or .isna().

Handle missing data by imputing with mean, median, or mode using SimpleImputer from sklearn, or remove rows with missing data using .dropna().

Data Transformation:

Convert categorical variables to numerical values using one-hot encoding or label encoding.

Normalize/standardize numerical columns using StandardScaler or MinMaxScaler from sklearn.

Subset and Filter Data:

Create subsets based on certain conditions (e.g., select rows where a specific feature value is greater than a threshold).

Filter outliers from numerical data using interquartile range (IQR).

Lab Assignment 2: Data Aggregation and Grouping with Pandas

Objective:
Master aggregation and grouping techniques using **Pandas** for summarizing data.

Tasks:

Group Data by Category:

Group data by one or more categorical features (e.g., "class" in the Iris dataset or "embarked" in Titanic dataset).

Use .groupby() to calculate aggregate statistics such as mean, median, sum, and count.

Pivot Tables:

Create a pivot table to summarize data (e.g., aggregate the average age of passengers in the Titanic dataset by class and gender).

Use .pivot_table() to perform multi-dimensional aggregation.

Data Aggregation and Custom Functions:

Apply custom aggregation functions to the grouped data (e.g., calculate custom metrics or perform complex transformations within each group).

Sorting and Ranking Data:

Sort the dataset by multiple columns (e.g., sorting by "age" or "fare").

Rank data based on specific metrics (e.g., assign ranks to passengers by fare in the Titanic dataset).

Lab Assignment 3: Data Visualization with Matplotlib and Seaborn

Objective:
Learn to visualize datasets using **Matplotlib** and **Seaborn** for better understanding and insights.

Tasks:

Basic Plotting with Matplotlib:

Create simple plots like line plots, bar plots, and histograms using **Matplotlib**.

Customize the plots by setting titles, labels, and legends.

Create scatter plots to visualize relationships between two variables.

Advanced Plotting with Seaborn:

Use **Seaborn** to create advanced visualizations like pair plots, heatmaps, box plots, and violin plots.

Customize visualizations with color palettes, styling, and themes.

Create a correlation heatmap to visualize correlations between features in the dataset.

Distribution Visualizations:

Plot distributions of continuous variables using **Seaborn's** distplot() or kdeplot().

Create bar plots for categorical variables to understand their frequency distribution.

Multi-Plot Grid Layouts:

Use **Matplotlib's** subplots() function to create multiple plots in a grid layout for comparison (e.g., scatter plot and histogram in the same figure).

Lab Assignment 4: Visualizing Relationships and Feature Importance

Objective:
Understand how to visualize relationships between features and evaluate feature importance for predictive models.

Tasks:

Scatter Plot Matrix:

Use **Seaborn's** pairplot() to create a scatter plot matrix to visualize the relationships between multiple features.

Analyze the pairwise relationships between features and identify any patterns or correlations.

Heatmap of Correlation Matrix:

Use **Pandas** to calculate the correlation matrix of numeric features.

Visualize the correlation matrix using **Seaborn's** heatmap() to understand feature correlations and multicollinearity.

Feature Importance from Models:

Train a decision tree or random forest model using **scikit-learn** on a dataset (e.g., **Iris** or **Titanic**).

Visualize feature importance using a bar chart to understand which features have the most impact on the model.

Visualizing Predictions vs. Actual Values:

For regression tasks, visualize the predicted values against the actual values using a scatter plot.

For classification tasks, visualize the classification results with a confusion matrix.

Lab Assignment 5: Time Series Data Visualization and Processing

Objective:
Learn how to process and visualize time series data, which is common in AI applications like forecasting and trend analysis.

Tasks:

Load and Preprocess Time Series Data:

Load a time series dataset (e.g., stock market data, weather data).

Parse dates properly and set the date column as the index using pd.to_datetime() and .set_index().

Plot Time Series Data:

Plot a time series line chart using **Matplotlib** to visualize trends over time.

Create rolling averages (e.g., 7-day, 30-day) to smooth out short-term fluctuations in the time series data.

Seasonal Decomposition of Time Series:

Use **statsmodels** to decompose a time series into seasonal, trend, and residual components.

Visualize the decomposed components to understand seasonal variations.

Forecasting with Simple Models:

Use simple forecasting models (e.g., moving average, ARIMA) to predict future values.

Visualize the forecasted data along with actual historical data.

Module 3	Introduction to Machine Learning	Assignments	Implementation	10 Sessions

Lab Assignment 3: Implementing Linear Regression

Tasks:

Load a real-world dataset (e.g., **Boston Housing Price** dataset).

Train a **Linear Regression** model using LinearRegression() from scikit-learn.

Evaluate the model using **Mean Squared Error (MSE)** and **R-squared Score**.

Visualize the regression line using Matplotlib.

Lab Assignment 4: Logistic Regression for Classification

Tasks:

Load the **Iris** or **Breast Cancer** dataset.

Preprocess the dataset (handle missing values, encode categorical variables, scale data).

Train a **Logistic Regression** model using LogisticRegression().

Evaluate performance using **Accuracy, Precision, Recall, F1-score**.

Plot the **Confusion Matrix** and **ROC Curve**.

Lab Assignment 5: Implementing K-Nearest Neighbors (KNN)

Tasks:

Load the **Iris dataset** and split it into training and testing sets.

Train a **KNN classifier** using KNeighborsClassifier().

Experiment with different values of **K** and evaluate performance.

Visualize decision boundaries using a **scatter plot**.

Lab Assignment 6: Decision Trees and Random Forests

Tasks:

Train a **Decision Tree classifier** on the Titanic dataset.

Visualize the tree structure using plot_tree().

Train a **Random Forest classifier** and compare performance with the decision tree.

Determine the **feature importance** using feature_importances_.

Module 4	Neural Networks and Deep Learning	Quiz	Implementation	10 Sessions

Lab Assignment 7: Introduction to Perceptron and Activation Functions

Tasks:

Implement a **single-layer perceptron** using NumPy.

Train the perceptron to classify **AND, OR, XOR** gates.

Experiment with different **activation functions** (Sigmoid, ReLU, Tanh).

Visualize decision boundaries.

Lab Assignment 8: Building a Simple Neural Network with Keras

Tasks:

Load the **MNIST dataset** from keras.datasets.

Preprocess the data (normalize pixel values, reshape input).

Create a **fully connected neural network** using Sequential API.

Train and evaluate the model using **categorical cross-entropy loss** and **accuracy**.

Lab Assignment 9: Implementing CNN from Scratch

Tasks:

Load the **CIFAR-10 dataset**.

Build a CNN with **Conv2D, MaxPooling2D, Flatten, Dense, Dropout** layers.

Use **Adam optimizer** and **categorical cross-entropy loss**.

Train and visualize loss/accuracy curves.

Lab Assignment 10: Image Augmentation & Regularization

Tasks:

Apply **data augmentation** (rotation, zoom, flipping) using ImageDataGenerator.

Add **dropout and batch normalization** to prevent overfitting.

Compare model performance with and without augmentation.

Lab Assignment 11: Transfer Learning with Pre-trained Models

Tasks:

Use **VGG16 or ResNet50** pre-trained on ImageNet.

Replace the output layer to classify **new images**.

Freeze earlier layers and fine-tune deeper layers.

Evaluate the model on a custom **dataset (e.g., Cats vs. Dogs)**.

Lab Assignment 12: Implementing RNN for Text Classification

Tasks:

Load **IMDB movie reviews dataset** from keras.datasets.

Preprocess text (tokenization, padding sequences).

Build an **RNN** with **Embedding, SimpleRNN, Dense** layers.

Train and evaluate the model.

Lab Assignment 13: Building an LSTM for Time Series Prediction

Tasks:

Load a **time series dataset** (e.g., stock prices, temperature data).

Preprocess the data (normalize, reshape).

Build an **LSTM-based model**.

Predict future values and visualize trends.

Targeted Application & Tools that can be used:

Applications:

Data Preprocessing: Clean and manipulate data from various sources such as CSV, Excel, SQL databases, and APIs.

Exploratory Data Analysis (EDA): Gain insights into datasets by identifying trends, patterns, and outliers.

Predictive Modeling: Build models for classification (e.g., spam detection) and regression (e.g., house price prediction).

Clustering: Group data into clusters for unsupervised learning tasks (e.g., customer segmentation).

Model Evaluation: Assess model performance using appropriate metrics such as accuracy, precision, recall, and F1-score.

Tools:

Pandas: For data manipulation and cleaning (e.g., handling missing values, merging datasets).

NumPy: For numerical operations and working with arrays and matrices.

Matplotlib: For creating static, animated, and interactive visualizations.

Seaborn: For advanced data visualizations (e.g., heatmaps, pair plots).

Plotly: For creating interactive visualizations, especially useful for large datasets.

Scikit-learn: The go-to library for implementing machine learning algorithms (e.g., linear regression, decision trees, k-means clustering).

XGBoost: For advanced gradient boosting models, particularly for large-scale machine learning tasks.

TensorFlow (for deep learning in Module 4): A powerful open-source library for building machine learning and deep learning models.

Keras: High-level neural network API, built on top of TensorFlow, to easily create deep learning models.

NLTK: The Natural Language Toolkit for various text processing tasks like tokenization, stemming, and part-of-speech tagging.

spaCy: A fast NLP library for advanced NLP tasks such as named entity recognition and dependency parsing.

Transformers (by Hugging Face): A powerful library for using pre-trained Transformer-based models like BERT, GPT, and others for advanced NLP tasks.

Text Book(s):

T1: ***Essentials of Python for Artificial Intelligence and Machine Learning*** *by Pramod Gupta and Anupam Bagchi*

Reference(s):

"Artificial Intelligence with Python" – Prateek Joshi

"Python Machine Learning" – Sebastian Raschka & Vahid Mirjalili

"Hands-On Artificial Intelligence with Python" – Teet Straus

"Deep Learning for Coders with Fastai and PyTorch" – Jeremy Howard & Sylvain Gugger

Course Code: **CSE2509**	Mobile Applications and Development Lab	**L- T- P- C**	0	0	4	2
Version No.	1.0					
Course Pre-requisites	The student needs to have fundamental understanding of object-oriented programming concepts with Java.					
Anti-requisites						
Course Description	The course deals with the basics of android platform and application life cycle. The goal of the course is to develop mobile applications with Android containing at least one of the following phone material components: GPS, accelerometer or phone camera, use simple GUI applications and work with database to store data locally or in a server. Topics include user interface design; user interface building; input methods; data handling; network techniques and URL loading; GPS and motion sensing. Android application framework and deployment. Power management, Screen resolution, Touch interface, Store data on the device.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Mobile Applications and Development** as mentioned above and attain **Employability Skills** through **Experiential Learning** Techniques.					
Course Out Comes	On successful completion of the course the students shall be able to: 1. Discuss the fundamentals of mobile application development and its architecture. (Understand) 2. Illustrate mobile applications with appropriate android view. (Apply) 3. Demonstrate the use of services, broadcast receiver, Notifications and content provider (Apply) 4. Apply data persistence techniques, to perform CRUD operations. (Apply) 5. Use multimedia and internet services for mobile applications. (Apply)					
Course Content:						

List of Laboratory Tasks

1.a. Design an app to read user inputs using edit text and display the result of arithmetic operations using toast message.

1.b. Create an android app to calculate the current age of yourself, select your DOB using date picker.

2.a. Design an app to input your personal information. Use autocomplete text view to select your place of birth.

2.b. Design an app to select elective course using spinner view and on click of the display button, toast your ID and selected elective course.

3. Design a restaurant menu app to print the total amount of orders.

4. Develop an android app that uses intent to maintain the following scenario.

Check the eligibility criteria for voting. Input the Aadhar no., Name & age in the first activity. If the age is above 18, display the voter's detail in the second activity. Else, display, “You are not eligible to vote” in the second Activity.

5. Demonstrate the use of fragment with list of buttons representing various colors, and on click of these buttons, the appropriate color is filled in the next fragment.

Create an Android application to input the vitals of a person (temperature, BP). If the vitals are abnormal, give proper notification to the user.

6. Create an android app to for movie ticket booking. Save the user name of the customer using shared preferences. After completion of booking, retrieve the username from the shared preferences and print the ticket details.

7. Create an android application to manage the details of students' database using SQLite.Use necessary UI components, which perform the operations such as insertion, modification, removal and view.Presidency University needs an APP for Admission eligibility checking for students, for that you need to take the following information from the Student: registration ID, physics, chemistry and mathematics marks (PCM), fees is allotted as below criteria.

PCM (Total marks %)	Fee concession
90 above	80 %
70 to 89	60 %
Below 69 %	no concession

On click on the button “Registration” details should be stored in the database using SQLite. Create button DISPLAY ALL (full students list) on click on the button it should display the students list per the fee concession.

8. A company need to design an app that plays soft music automatically in the background. Create an app to achieve this functionality.

9. Create an android application such that your view object in the Activity can be Animated with fade-in effect. Create an appropriate XML file named fade-in and write the application to perform the property animation.

10. Demonstrate how to send SMS and email.

11. Create an android application to transfer a file using WiFi. Create an android application “Where am I” with an Activity that uses the GPS Location provider to find the device’s last known location.

Targeted Application & Tools that can be used:

Android Studio

Text Book

T1. Dawn Griffiths, David Griffiths, "Head First Android Development", OReilly, 3rd Edition, November 2021

References

R1. Barry Burd, "Android Application Development" All-in-one Dummies, Wiley, 3rd Edition, January 2021

R2. J F DiMarzio, "Beginning Android Programming with Android Studio", 4th Edition, Wiley, 2016.

R3. Pradeep kothari, "Android Application Development - Black Book", DreamTech Press, May 2014

R4. Erik Hellman, "Android Programming – Pushing the Limits", 1st Edition, Wiley, 2014.

R5. Anubhav Pradhan, Anil V Deshpande, " Composing Mobile Apps" using Android, Wiley, 2014

E-Resources: https://puniversity.informaticsglobal.com/login Or http://182.72.188.193/

Topics relevant to the development of SKILLS: Graphics and Animation, App Widgets Skill Development through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE7000	**Course Title: Internship** **Type of Course:**	**L- T-P- C**	-	-	-	2
Version No.	**1.0**					
Course Pre-requisites	**Knowledge and Skills related to all the courses studied in previous semesters.**					
Anti-requisites	**NIL**					
Course Description	Students observe science and technology in action, develop an awareness of the method of scientific experimentation, and often get an opportunity to see, study and operate sophisticated and costly equipment. They also learn about the implementation of the principles of management they have learnt in class, when they observe multidisciplinary teams of experts from engineering, science, economics, operations research, and management deal with techno-economic problems at the micro and macro levels. Finally, it enables them to develop and refine their language, communication and inter-personal skills, both by its very nature, and by the various evaluation components, such as seminar, group discussion, project report preparation, etc. The broad-based core education, strong in mathematics and science and rich in analytical tools, provides the foundation necessary for the student to understand properly the nature of real-life problems. The students have options to pursue this course as either Project Work and Dissertation at the university, or Project Work in an Industry/ Company/ Research Laboratory, or Internship Program in an Industry/Company.					
Course Objectives	The objective of the course is to familiarize the learners with the concepts of Professional Practice and attain Employability Skills through Experiential Learning techniques.					
Course Outcomes	On successful completion of this course the students shall be able to: tify the engineering problems related to local, regional, national or global needs. ly appropriate techniques or modern tools for solving the intended problem. gn the experiments as per the standards and specifications. pret the events and results for meaningful conclusions. the project development on GitHub raise project findings and communicate effectively through scholarly publications.					

Course Code: **CSE2506**	**Course Title:** Cloud computing **Type of Course:** Theory		**L- T-P- C**	2	**0**	0	**2**
Version No.	2.0					
Course Pre-requisites	CSE1507 Data Communication and Computer Networks					
Anti-requisites	**NIL**					
Course Description	This Course is designed to introduce the concepts of Cloud Computing as a new computing paradigm. Cloud Computing has emerged in recent years as a new paradigm for hosting and delivering services over the Internet. The students can explore various Cloud Computing terminology, principles and applications. Understanding different views of the Cloud Computing such as theoretical, technical and commercial aspects. Topics include: Evolution of cloud computing and its services available today, Introduction, Architecture of cloud computing, Infrastructure, platform, software, Types of cloud, Business models, cloud services, Collaborating using cloud services, Virtualization for cloud, Security, Standards and Applications.						
Course Objective	The objective of the course is to familiarize the learners with the concepts of Could computing and Virtualization and attain Employability through Participative Learning techniques.						
Course Outcomes	On successful completion of the course the students shall be able to: · Describe fundamentals of cloud computing, virtualization and cloud computing services. · Discuss high-throughput and data-intensive computing. · Explain security and standards in cloud computing. · Demonstrate the installation and configuration of virtual machine.						
Course Content:							
Module 1	Introduction to Cloud and Virtualization	Assignment	Virtualization			**12 Sessions**	
Topics: Introduction to Cloud and Virtualization Cloud Computing at a Glance, Historical Developments, Building Cloud Computing Environments, Computing Platforms and Technologies, Virtualization, Characteristics of Virtualized Environments Taxonomy of Virtualization Techniques, Virtualization and Cloud Computing, Technology Examples, Cloud Computing Architecture, IaaS, PaaS, SaaS, Types of Clouds, Economics of Cloud							

Module 2	High Throughput and Data Intensive Computing	Assignment	Virtualization	**12 Sessions**

Topics:

High Throughput and Data Intensive Computing: Task computing, MPI applications, Task based programming, Introduction to DIC, Technologies for DIC, Aneka Map Reduce Programming.

Module 3	Cloud Security and Standards	Assignment	Virtualization	**10 Sessions**

Topics:

Cloud Security and Standards: Cloud Security Challenges, Software-as-a-Service Security, Application standards, Client standards, Infrastructure and Service standards.

Module 4	Cloud Platforms	Assignment	Virtualization	**11 Sessions**

Cloud Platforms, Advances in cloud: introduction to Amazon Web Services: Introduction to Google App Engine, Introduction to Microsoft Azure. Media Clouds - Security Clouds - Computing Clouds - Mobile Clouds – Federated Clouds – Hybrid Cloud

Targeted Application & Tools that can be used:

Text Book(s):

1. John Rittinghouse and James Ransome, “Cloud Computing, Implementation, Management and Security”, CRC Press.

2. Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi, “Mastering Cloud Computing”, McGraw Hill Education.

Reference(s):

1. David E.Y. Sarna, “Implementing and Developing Cloud Applications”, CRC Press.

2. Anthony T Velte, Toby J Velte, Robert Elsenpeter, “Cloud Computing: A Practical Approach”, Tata McGraw-Hill. Web resources: https://presiuniv.knimbus.com/user#/home

Course Code: CSE2514	**Compiler Design** **Type of Course: Theory Only**	**L-T-P-C**	3	0	0	3
Version No.	2.0					
Course Pre-requisites	nil					
Anti-requisites	NIL					
Course Description	The Course is intended to teach the students the basic techniques that underlie the practice of Compiler Construction. The Course will introduce the theory and tools that can be employed in order to perform syntax-directed translation of a high-level programming language into an executable code. Topics consist of: Introduction to Compilers, Language translators: compilersand interpreters. Lexical Analysis, Role of the parser ,semantic analysis, Intermediate Code Generation, Code Optimization, DAG representation of Basic Blocks, Global optimization, Peephole Optimization, Garbage Collection, Parallel Architectures.					
Course Objective	**The objective of the course is to familiarize the learners with the concepts of Compiler Designand attain SKILL DEVELOPMENT through PARTICIPATIVE LEARNING techniques.**					
Course Out Comes	On successful completion of the course the students shall be able to: 1. Explain the basic concepts of compiler and its various phases. 2. Construct front end of the compiler. 3. Apply suitable data structure to improve efficiency of compiler. 4. Generate Intermediate code for the given statements. 5. Discuss how to optimize the program for backend of the compiler for different computer architecture					
Course Content:						
Module 1	Introduction And LexicalAnalysis	Term paper	Data Analysis	**13 Sessions**		
Topics: Compilers , Analysis of the source program ,Phases of a compiler ,Cousins of the Compiler , Grouping of Phases, Compiler construction tools , Lexical Analysis , Role of Lexical Analyzer , Input Buffering, Specification of Token, – Recognizer - Introduction to LEX Programming.						
Module 2	Syntax Analysis	Term paper	Data Analysis	**15 Sessions**		
Topics: Role of the parser, Top Down parsing, Recursive decent parser - Predictive parser -Bottom-up parsing Shift reduce parser - LR parser – SLR parser – Canonical parser – LALR parser - YACC programming.						
Module 3	Semantic Analysis AndIntermediate Code Generation	Data Analysis	Data Analysis	**8 Sessions**		
Introduction to syntax directed translation - Synthesis and inherited attributes - Type Checking - Type Conversions .Topics: Intermediate languages, Declarations, Assignment Statements , Boolean Expressions ,Case Statements –Back patching – Looping statements - Procedure calls.						
Module 4	Code Optimization	Data Analysis	Data Analysis	**8 Sessions**		
Topics: Optimization of basic Blocks, Introduction to Global Data Flow Analysis, Basic Blocks and Flow Graphs, Next-use Information, Machine Independent Code Optimizations, DAG representation of Basic Blocks, Peephole Optimization.						
Module 5	Code Generation	Data Analysis	Data Analysis	**8 Sessions**		

Storage Organization, Stack Allocation Space, Access to Non-local Data on the Stack, Heap Management, Issues in the design of code generator, The target machine Register allocation, A simple Code generator

Targeted Application & Tools that can be used:

The knowledge of this course can be applied in the building automatic translators (compilers) for higher levelprogramming languages. Professionally used software –lex and YACC

Assignment:

Assignment 1- Translate the arithmetic expression: a+ -(b+c) into quadraples, triples and indirect triples.Assignment 2- Draw the DAG for the arithmetic expressiona+a*(b-c)+(b-c)*d.

Text Book

1. Alfred V. Aho, Jeffrey D Ullman, “Compilers: Principles, Techniques and Tools”, Pearson .

References

1. Jean Paul Tremblay, Paul G Serenson, "The Theory and Practice of Compiler Writing", BS Publications.
2. C. N. Fischer and R. J. LeBlanc, “Crafting a compiler with C”, Benjamin Cummings.
3. HenkAlblas and Albert Nymeyer, “Practice and Principles of Compiler Building with C”, PHI.
4. Kenneth C. Louden, “Compiler Construction: Principles and Practice”, Thompson Learning.
5. Dhamdhere, D. M., "Compiler Construction Principles and Practice", Macmillan India Ltd.

E-Resources

https://puniversity.informaticsglobal.com:2229/login.aspx

Topics relevant to the development of SKILLS:

To optimize the program for backend of the compiler for different computer architecture for Skill Developmentthrough Participative Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE2503	**Cryptography and Network Security**		**L-T- P-C**	3	0	0	3
Version No.	2.0					
Course Pre-requisites	Basic Knowledge in Number Theory, Binary Operations					
Anti-requisites	**NIL**					
Course Description	The Course deals with the principles and practice of cryptography and network security, focusing in particular on the security aspects of the web and Internet.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Cryptography and Network Security above** and attain **Skill Development** through **Problem Solving** methodologies.					
Course Outcomes	On successful completion of this course the students shall be able to: 1. Describe the basic concept of Cryptography 2. Classify different types of Cryptographic Algorithms 3. Solve Mathematical problems required for Cryptography 4. Illustrate Network Security concepts					
Course Content:						
Module 1	**Introduction to Cryptography**	Assignment	Recognize the techniques			07 Sessions
Topics: Introduction to Cryptography, Model of Network Security, OSI Security architecture, Security Attacks: active attacks, passive attacks, services: Authentication, Access Control, Data Confidentiality, Data Integrity, Nonrepudiation, Substitution Ciphers : Play-fair and Hill Cipher, Vigenere cipher, Introduction to Block Cipher and Stream Cipher, Feistel Structure, ECB modes of block cipher						
Module 2	**Symmetric Encryption Algorithms**	Assignment	Analysis of results			09 Sessions
Topics: Symmetric Encryption Algorithms : Data Encryption Standard, Introduction to Galois Field, Advanced Encryption Standard, Modular Arithmetic, Prime numbers, Fermat's little theorem, Applications of Fermat's little theorem in modular athematic, brief about primality testing and factorization, Euclidean and Extended Euclidean Algorithm, Euler Totient Function, Chinese remainder theorem.						
Module 3	**Public Key Cryptography**	Assignment	Analysis of solutions			09 Sessions
Topics: Overview of Public Key Cryptography, RSA, Diffie-Helman Key exchange, Man in the middle attack, Cryptographic Hash functions, Secure Hash Algorithm, Message Authentication Codes – HMAC, Digital Signature, Ei-gamal Encryption, Elliptic curve cryptography overview						
Module 4	**Network Security**	Assignment	Analysis of solutions			05 Sessions

Topics:
Network Security fundamentals, Network Security applications: Authentication: Kerberos, PKI, Network Security applications: e-mail security: PGP, MIME, Network Security applications: IP Security: IPSec architecture, Network
Security applications: DNS Security.

Targeted Application & Tools that can be used:
Students get the knowledge about cryptography techniques followed, the algorithms used for encryption anddecryptions & the techniques for authentication and confidentiality of messages.

Textbooks:
T1 William Stallings, "Cryptography and Network Security - Principles and Practices", 7th Edition, Pearson publication, ISBN: 978-93-325-8522-5, 2017

References:
R1 Bruice Schneier, "Applied Cryptography – Protocols, Algorithms and Source code in C", Second Edition, Wiley Publication, ISBN: 978-81-265-1368-0, 2017
R2 Cryptography and Network Security, Express Learning, ITL Education Solution Limited.
R3 e-pg pathshala UGC lecture series

Web
references:
https://puniversity.informaticsglobal.com:2229/login.aspx?direct=true&db=nlebk&AN=2233842&site
=ehost-live
https://nptel.ac.in/courses/106105031.

Topics relevant to "Skill Development": Topics relevant to "Skill Development":

1. Play-fair and Hill Cipher
2. Euclidean and Extended Euclidean Algorithm
3. Secure Hash Algorithm
4. Diffie-Helman Key exchange
5. Totient Function.
6. Fermat's little theorem

Course Code: CSE1504	**Course Title: Web Technologies** **Type of Course: Program core** **Theory Only**		**L- T-P-C**	2-0-0-2
Version No.	2.0			
Course Pre-requisites	**NIL**			
Anti-requisites	**NIL**			
Course Description	This course highlights the basic web design using Hypertext Markup Language and Cascading Style Sheets. Students will be trained in planning and designing effective web pages by writing code using current leading trends in the web domain, enhancing web pages with the use of page layout techniques, text formatting, graphics, images, and multimedia. The focus is on popular key technologies that will help students to build Internet- and web-based applications that interact with other applications and with databases.			
Course Objective	**The objective of the course is to familiarize the learners with the concepts of Web Technology and attain Skill Development through Experiential Learning techniques.**			
Course Outcomes	**On successful completion of this course the students shall be able to:** **CO1:** Implement web-based application using client-side scripting languages. **(Application level)** **CO2**: Apply various constructs to enhance the appearance of a website. **(Application level)** **CO3**: Illustrate java-script concepts to demonstration dynamic web site**(Application level)** **CO4:** Apply server-side scripting languages to develop a web page linked to a database. **(Application level)**			
Course Content:				
Module 1	**Introduction to XHTML**	Quizzes and Assignments	Quizzes on various features of XHTML, simple applications	**8 Sessions**
Topics: Basics: Web, WWW, Web browsers, Web servers, Internet. XHTML: Origins and Evolution of HTML and XHTML: Basic Syntax, Standard XHTML Document Structure, Basic Text Markup, Images, Hypertext Links, Lists, Tables, Forms, Frames, Syntactic Differences between HTML and XHTML.				
Module 2	**Advanced CSS**	Quizzes and assignments	Comprehension based Quizzes and assignments; Application of CSS in designing webpages	**8 Sessions**
Topics: **CSS:** Introduction to CSS, Defining & Applying a style, Creating style sheets, types of style sheet, selectors, CSS font properties, border properties, Box model, opacity, CSS pseudo class and pseudo-elements. **Advanced CSS:** Layout, Normal Flow, Positioning Elements, Floating Elements, Responsive Design, CSS Frameworks **XML:** Basics, demonstration of applications using XML				
Module 3	**Fundamentals of JavaScript**	Quizzes and assignments	Application of JavaScript for dynamic web page designing	**7 Sessions**

Topics: JavaScript: Introduction to JavaScript, Basic JavaScript Instructions, Functions, Methods & Objects, Decisions and Loops, Document Object Model, Event handling, handling window pop-ups, JavaScript validation.				
Module 4	**PHP – Application Level**	Quizzes and assignments	Application of PHP in web designing	**7 Sessions**
Topics: **PHP:** Introduction to server-side Development with PHP, Arrays, $GET and $ POST, $_Files Array, Reading/Writing Files, PHP Classes and Objects, Working with Databases, SQL, Database APIs, Managing a MySQL Database. Accessing MySQL in PHP.				
Targeted Application & Tools that can be used: **Xampp web server to be used to demonstrate PHP.**				
Project work/Assignment:				
Assignments are given after completion of each module which the student need to submit within the stipulated deadline.				
Textbook(s): 1] Robert. W. Sebesta, "*Programming the World Wide Web"*, Pearson Education, 8th Edition, 2015. 2] *CSS Notes for Professionals*, ebook available at https://books.goalkicker.com/CSSBook/ (Retrieved on Jan. 20, 2022) 3] Deitel, Deitel, Goldberg,"*Internet & World Wide Web How to Program*", Fifth Edition, Pearson Education, 2021.				
References 1] Randy Connolly, Ricardo Hoar, "Fundamentals of Web Development", Pearson Education India, 1st. Edition.2016. 2] Jeffrey C. Jackson, "Web Technologies: A Computer Science Perspective", Pearson Education, 1st Edition,2016.				
Topics related to development of "FOUNDATION": Web, WWW, Web browsers, Web servers, Internet. CSS, PHP. Designing for healthcare. **for Skill Development through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.** **E-References** **pu.informatics.global, https://sm-nitk.vlabs.ac.in/**				

urse Code: **E2510**	**urse Title:** Competitive Programming and Problem Solving **pe of Course: Program Core**	-P-C	0	0		

rsion No.	
urse Pre-requisites	
ti-requisites	
urse Description	The **Competitive Programming and Problem Solving** course equips students with efficient problem-solving skills for coding competitions and real-world challenges. Starting with brute-force solutions, students learn to optimize time and space complexity using advanced techniques like dynamic programming, greedy algorithms, and backtracking. Hands-on practice on platforms like CodeChef and Codeforces helps tackle problems involving number theory, data structures, and algorithmic paradigms. By understanding CP constraints and fostering a strategic mindset, students gain the confidence to excel in competitions, technical interviews, and practical applications.
urse Out Comes	On successful completion of the course the students shall be able to: CO1 : Understanding the issues of online platforms and Competitive Programming (CP) and developing brute force coding for commonly asked CP problems. CO2 : Analyzing the space and time complexity of brute force solutions and designing efficient solutions. CO3 : Evaluating the applicability of suitable algorithmic approaches to solve relevant CP problems. CO4: Creating efficient solutions of CP problems using the learnt algorithmic approaches.
urse Objective	The objective of the course is to familiarize the learners with the concepts of **Competitive Programming and Problem Solving** and attain **Skill Development** through **Experiential Learning** techniques.

dule 1: Introduction to Competitive Programming

erview of Efficient Coding for Problem Solving and CP: Introduction to competitive programming (CP); revisit of complexity analysis; introduction to online platforms such as codechef, codeforces etc and online submission; constraints during CP, online testing process and common errors such as TLE; use of STL

dule 2: Number Theory for Problem-Solving

e of Number Theory for problem-solving: reducing time/space complexity of brute force coding solution of Sieve Method, Inverse Module, Euclidian Method of factorization; efficient coding for Permutation Combination; XORing based and pattern-based solutions.

dule 3: Optimizing Time & Space Using Sequential Storage

ding for Optimizing time and Space using Sequential Storage: two pointer approach; problem-solving using arrays and strings such as rotation on sorted arrays, duplicate removal, string matching algorithms; Kadane's algo, stacks, priority-queues and hashing based efficient coding; median based problems and alternate solutions.

dule 4: Non-Linear Data Structures

plying Non-Linear Data Structures for real-life problems: design of efficient solutions for problems such as finding loops in a linked list, memory efficient DLL, block reversal in LL; problem solving using trees and binary trees, Catalan numbers, applications of graphs, spanning tree and path algos for CP problems with reduced time/space complexity.

dule 5: Problem Solving using Advanced Topics

Problem Solving using Advanced Topics: concept of disjoint sets and their efficient representation, algorithmic approaches such as Greedy, Backtracking, Dynamic Programming and applying them for CP problems using bottom-up dynamic programming.

: of Laboratory Tasks:

1. You are given the finishing times of 'N' runners in a marathon. Write a program to find the runner who finished in the third position. **Focus:** Basic data structures (arrays), sorting algorithms (e.g., insertion sort, selection sort), and basic input/output.
2. In the same marathon, you are given the finishing times of 'N' runners and their bib numbers. Write a program to efficiently find the top 10 runners and their corresponding bib numbers. **Focus:** Efficient sorting algorithms (e.g., merge sort, quick sort), data structures like priority queues, and optimizing for large datasets.
3. A library maintains a list of books with their unique IDs. Write a program to check if a given book ID is present in the library. Focus: Searching algorithms (linear search), basic data structures (arrays or lists).
4. The library wants to implement a system to quickly find books by their titles. Suggest an efficient data structure (e.g., a hash table or a trie) and explain how to implement it to achieve fast book lookups. Focus: Understanding the trade-offs between different data structures, choosing the most appropriate data structure for a specific problem, and implementing efficient search operations.
5. An online store sells products with different prices. Write a program to calculate the total cost of a given list of products. **Focus:** Basic arithmetic operations, working with arrays or lists to store product prices.
6. The online store offers discounts based on the total purchase amount. Design an algorithm to efficiently calculate the final cost of an order, considering different discount rules (e.g., percentage discounts, fixed amount discounts, tiered discounts). Focus: Algorithmic design, conditional statements, handling complex scenarios with multiple rules, and potentially using dynamic programming techniques for optimization.
7. You are given two integers, 'a' and 'm'. Calculate 'a' raised to the power 'm' modulo a large prime number 'p'. **Focus:** Basic modular arithmetic operations (modular exponentiation), understanding the modulo operator.
8. In a secure communication system, you need to efficiently compute the modular exponentiation for very large values of 'm'. Implement and analyze the efficiency of the binary exponentiation algorithm for this task. **Focus:** Efficient algorithms for modular exponentiation (binary exponentiation), time complexity analysis, and understanding the importance of efficient algorithms in cryptography.

9. You have a deck of 'N' cards. Calculate the total number of possible hands of size 'K' that can be drawn from the deck. **Focus:** Basic combinatorics (combinations), factorial calculations.
10. In a card game, you need to calculate the probability of drawing certain combinations of cards (e.g., a pair, a three-of-a-kind) from a shuffled deck. Design an efficient algorithm to calculate these probabilities. **Focus:** Advanced combinatorics (permutations and combinations with repetitions), probability calculations, and optimizing calculations to avoid overflows.
11. You are given a network of devices represented as a graph. Determine if there is a path between two given devices in the network. **Focus:** Graph traversal algorithms (depth-first search or breadth-first search).
12. In a secure network, you need to detect and isolate compromised devices. Design an algorithm that efficiently identifies devices that exhibit anomalous behavior (e.g., unusual traffic patterns) using XOR-based techniques for data comparison and pattern matching. **Focus:** Applying XOR operations for data comparison and pattern recognition, understanding the properties of XOR (e.g., commutative, associative), and designing algorithms for network anomaly detection.
13. You are given an array representing the speeds of cars on a highway. Find the minimum time required for all cars to pass a certain point. **Focus:** Basic array traversal, finding the minimum element in an array.
14. In a more realistic scenario, cars have different lengths. Implement a two-pointer approach to simulate the movement of cars and determine the minimum time for all cars to pass a given point. **Focus:** Two-pointer technique, simulating real-world scenarios with arrays, optimizing time complexity.
15. Given a string, find the number of occurrences of a specific substring within the string. **Focus:** Basic string manipulation, string matching (brute-force approach).
16. Implement the KMP (Knuth-Morris-Pratt) string matching algorithm to efficiently find all occurrences of a given pattern within a large text document. **Focus:** Advanced string matching algorithms, understanding the concept of the "next" array in KMP, optimizing for large input sizes.
17. An online auction platform receives bids for different items. Implement a data structure (e.g., a priority queue) to efficiently track the highest bid for each item. **Focus:** Priority queues, insertion and extraction operations on priority queues, basic implementation of a priority queue using an array or a suitable library.
18. The auction platform needs to handle a large number of bids concurrently. Design and implement a system that efficiently processes bids, updates the highest bid for each item, and handles potential race conditions. **Focus:** Concurrent data structures and algorithms, thread safety, handling race conditions, optimizing for high-throughput scenarios.
19. A social network can be represented as a graph where users are nodes, and connections between users are edges. Write an algorithm to find if two given users are connected in the network. **Focus:** Graph traversal algorithms (depth-first search or breadth-first search), basic graph representation (adjacency list or adjacency matrix).
20. In a large social network, efficiently finding the shortest path between two users is crucial. Implement Dijkstra's algorithm to find the shortest paths between users in the network, considering edge weights (e.g., representing the strength of connections). **Focus:** Shortest path algorithms (Dijkstra's algorithm), graph algorithms with weighted edges, optimizing for large graphs.
21. A file system can be modeled as a tree structure. Implement a function to traverse the file system and print the names of all files and directories. **Focus:** Tree traversal algorithms (depth-first search or breadth-first search), basic tree representation (using nodes and pointers).
22. Design and implement a file system that supports efficient operations like creating directories, deleting files, and finding files based on their names or paths. Consider using a combination of tree structures and

hash tables for efficient indexing and searching. **Focus:** Designing and implementing file system structures, using multiple data structures together, optimizing for common file system operations.

23. An online shopping cart can be represented as a tree, where each node represents an item or a category of items. Write an algorithm to calculate the total price of all items in the shopping cart. **Focus:** Tree traversal, calculating sums within a tree structure.
24. Implement a system that allows customers to apply discounts and coupons to their shopping carts. Consider using a combination of trees and other data structures (e.g., hash tables) to efficiently apply discounts and calculate the final price. **Focus:** Applying discounts and promotions to tree-like structures, efficient implementation of discount rules, optimizing for complex pricing scenarios.
25. In a social network, users can form groups. Given a list of friendships, determine if all users in a specific group are connected (directly or indirectly) through friendships. **Focus:** Disjoint set union (DSU) data structure, basic connectivity checks.
26. Design an efficient algorithm to find the minimum number of new friendships needed to connect all users in the social network into a single, connected component. **Focus:** Applying DSU for finding connected components, greedy algorithms, optimization for minimizing connections.
27. A treasure hunt involves a series of clues leading to the final treasure. Given a list of possible paths and their associated costs, find the cheapest path to reach the treasure. **Focus:** Greedy algorithms (e.g., Dijkstra's algorithm for shortest paths), basic graph representation.
28. In a more complex treasure hunt, there are time constraints associated with each path. Design an algorithm to find the fastest path to the treasure while considering both path costs and time constraints.

Focus: Combining greedy approaches with other techniques (e.g., priority queues), handling multiple constraints, optimizing for time-critical scenarios.

29. In a simplified chess game with only rooks, determine the minimum number of moves required for a rook to reach a specific target square on an empty board. **Focus:** Breadth-first search (BFS) on a graph (the chessboard), basic graph traversal.
30. In a more realistic chess game with multiple pieces and obstacles, implement a minimax algorithm with alpha-beta pruning to determine the best move for a player. **Focus:** Game tree search, minimax algorithm, optimization techniques like alpha-beta pruning, handling complex game states.

geted Application & Tools that can be used:

1. C or C++ Compiler (g++): The standard compiler for CP. Familiarize students with compilation flags (e.g., -O2 for optimization).
2. IDE (Integrated Development Environment): Code:: Blocks, Visual Studio, CLion, or similar IDEs. These provide debugging capabilities, code completion, and other helpful features.
3. Online Judges (CodeChef, Codeforces, LeetCode, HackerRank): Essential for practicing and submitting solutions.
4. Debugger (gdb): Crucial for understanding code execution and finding bugs. Origin, excel and Mat lab soft wares for programming and data analysis.
5. Number Theory Libraries: Some libraries provide pre-built functions for number theory operations (though often it's better to implement them yourself for learning).
6. Wolfram Alpha: A useful tool for verifying number theory calculations and exploring concepts.

7. **String Libraries:** Familiarize students with the string manipulation functions available in C++.
8. **Graph Visualization Tools:** Tools like Graphviz can be helpful for visualizing graphs and understanding graph algorithms.
9. **DP Debugging Techniques:** Practice debugging DP solutions, as they can be complex. Visualizing the DP table can be helpful.

xt Books:

1 **Guide to Competitive Programming: Learning and Improving Algorithms Through Contests" (3rd Edition),** ***Antti Laaksonen, springer, 2024***

2 **"Data Structures and Algorithms in Java: A Project-Based Approach" –** ***Dan S. Myers, Cambridge University Press***

Reference Books:

1. **Data Structures and Algorithmic Thinking with Python/C++/Java",** ***Narasimha Karumanchi, 5th Edition, Career Monk, 2017.***
2. **Introduction to Algorithms,** Thomas H. Cormen **(Author),** Charles E. Leiserson **(Author),** Ronald L. Rivest **, fourth edition April 2022**

Web Resources

1. https://nptel.ac.in/courses/106106231
2.

ject work/Assignment: Mention the Type of Project /Assignment proposed for this course

Assessment Type

- Midterm exam
- Assignment (review of digital/ e-resource from PU link given in references section - mandatory to submit screen shot accessing digital resource.)
- Quiz
- End Term Exam
- Self-Learning

Catalogue prepared by	Dr. Robin Rohit Vincent
Recommended by the Board of Studies on	
Date of Approval by the Academic Council	

Course Code: **CSE2507**	**Course Title:** Cloud computing **Type of Course** Lab		**L- T-P- C**	0	**0**	2	**1**
Version No.	1.0					
Course Pre-requisites	CSE1507 Data Communication and Computer Networks					
Anti-requisites	**NIL**					
Course Description	This course is designed to give hands-on experience with cloud platforms, services, and deployment models. Students will learn to set up, configure, and manage cloud environments using platforms like AWS, Microsoft Azure, and Google Cloud. The course covers virtualization, containerization, serverless computing, cloud storage, security, and scalability. Through practical assignments, students will develop skills in deploying cloud applications, managing cloud resources, automating cloud workflows, and implementing cost-effective cloud solutions.					
Course Objective	The objective of the course is to Understand Cloud Infrastructure, deploy and manage virtual machines, implement cloud storage, develop and deploy cloud applications, optimize cost and performance.					
Course Outcomes	On successful completion of the course the students shall be able to: CO1· Deploy and Manage Cloud Resources. CO2. Develop and Deploy Cloud-based Applications CO3. Optimize Performance and Cost in the Cloud CO4. Implement Security and Automation in Cloud Environments					
Course Content:						
Module 1	Introduction to Cloud and Virtualization	Assignment	Virtualization			**4 Sessions**

Lab Assignment 1: Setting Up Virtual Machines on Cloud

- Create a **Virtual Machine (VM)** on AWS/Azure/GCP
  Configure OS, storage, and network settings
- Connect to the VM using SSH/RDP
  Install web server (Apache/Nginx) and deploy a static webpage

Lab Assignment 2: Containerization Using Docker

- Install **Docker** on a local or cloud VM
- Create and run a **Docker container**

- Build a **custom Docker image** with a simple Python/Node.js application
- Push the image to **Docker Hub** and deploy it on a new VM

Module 2	High Throughput and Data Intensive Computing	Assignment	Virtualization	**9 Sessions**

Lab Assignment 1: Setting Up a Distributed Computing Environment

- Launch a **Hadoop or Spark cluster** on AWS EMR / Azure HDInsight / Google Dataproc
- Configure **HDFS (Hadoop Distributed File System)** for big data storage
- Run a **basic MapReduce job** on sample data

Lab Assignment 2: Data Preprocessing with Cloud Storage

- Store large datasets in **Amazon S3 / Azure Blob Storage / Google Cloud Storage**
- Use **Apache Spark or Hadoop** to read, clean, and process data
- Convert datasets into **Parquet or Avro formats** for efficient storage

Lab Assignment 3: Batch Processing with Apache Spark

- Load large datasets (e.g., logs, tweets, transaction data) into **Spark DataFrame**
- Perform **ETL (Extract, Transform, Load) operations** on the data
- Use **SparkSQL** for querying large datasets

Lab Assignment 4: Real-Time Data Processing with Spark Streaming

- Set up **Kafka / AWS Kinesis / Google Pub/Sub** for real-time data ingestion
- Process streaming data using **Spark Streaming**
- Perform windowed aggregations and visualize real-time trends

Lab Assignment 5: Cloud-Based Machine Learning with Big Data

- Use **Google BigQuery ML / AWS SageMaker / Azure Machine Learning** for model training
- Train a **linear regression or classification model** on a large dataset
- Deploy the trained model as an API for real-time predictions

Lab Assignment 6: Running Parallel Machine Learning Workloads

- Implement **distributed ML training** using **Spark MLlib or TensorFlow on Cloud TPUs**
- Train models on a large dataset and optimize performance using distributed execution

Lab Assignment 7: Auto-Scaling and Load Balancing for Data Processing

- Deploy a **serverless Spark job** using **AWS Glue / Azure Synapse**
- Implement **auto-scaling for high-throughput jobs**
- Measure performance improvements using cloud monitoring tools

Lab Assignment 8: Cost Optimization for High-Throughput Data Processing

- Analyze **cloud cost reports** for data-intensive workloads
- Optimize cloud storage and compute resources for cost-efficiency
- Compare **on-demand vs. reserved vs. spot instances** for cost savings

Module 3	Cloud Security and Standards	Assignment	Virtualization	**9 Sessions**

Lab Assignment 9: Configuring Identity and Access Management (IAM)

- Set up **IAM roles and policies** in AWS / Azure / Google Cloud
- Create and assign users with **least privilege access**
- Implement **Multi-Factor Authentication (MFA)** for added security
- Audit IAM policies using **AWS IAM Access Analyzer / Azure Security Center**

Lab Assignment 10: Setting Up Single Sign-On (SSO) and Role-Based Access Control (RBAC)

- Configure **AWS Cognito / Azure Active Directory / Google IAM** for authentication
- Implement **Role-Based Access Control (RBAC)** for users and groups
- Integrate **OAuth 2.0 / OpenID Connect (OIDC) / SAML** for secure authentication

Lab Assignment 11: Encrypting Data at Rest and in Transit

- Encrypt **cloud storage (S3, Blob, Cloud Storage)** using **KMS (Key Management Service)**
- Set up **TLS/SSL certificates** for secure web traffic encryption
- Enable **database encryption (AWS RDS, Azure SQL, GCP Cloud SQL)**

Lab Assignment 12: Implementing Compliance & Governance in Cloud

- Enable **GDPR, HIPAA, ISO 27001 compliance tools** in cloud platforms
- Use **AWS Config / Azure Policy / GCP Security Command Center** to enforce compliance
- Conduct **security audits and generate compliance reports**

Lab Assignment 13: Implementing Cloud Monitoring & Threat Detection

- Configure **AWS CloudTrail / Azure Monitor / GCP Operations Suite** for activity logging
- Set up **intrusion detection systems (IDS) & anomaly detection**

- Analyze security logs using **Amazon GuardDuty / Azure Sentinel / Chronicle Security**

Lab Assignment 14: Automating Security Incident Response

- Deploy a **Serverless Lambda / Azure Logic App** to automatically respond to security incidents
- Implement **automated alerts** for suspicious activity
- Test a **denial-of-service (DDoS) simulation** and implement mitigation strategies

Module 4	Cloud Platforms	Assignment	Virtualization	**8 Sessions**

Lab Assignment 15: Getting Started with Cloud Platforms

- Create a **free-tier account** on AWS, Azure, or Google Cloud
- Navigate the **Cloud Console, CLI, and SDKs**
- Explore and configure **dashboard, billing, and IAM settings**

Lab Assignment 16: Launching a Virtual Machine (VM) on Cloud

- Deploy a **VM instance** using **AWS EC2, Azure Virtual Machines, or Google Compute Engine**
  Configure **OS, storage, networking, and security groups**
  Connect to the instance using **SSH (Linux) or RDP (Windows)**

Lab Assignment 17: Cloud Storage and File Management

- Create **Object Storage (AWS S3 / Azure Blob Storage / Google Cloud Storage)**
  Upload, download, and set access permissions for files
  Implement **Lifecycle Policies and Versioning**

Lab Assignment 18: Cloud Database Management

- Deploy a **Relational Database (AWS RDS / Azure SQL Database / Cloud SQL)**
  Connect and query the database using **MySQL/PostgreSQL clients**
  Set up database **backups and automatic scaling**

Lab Assignment 19: Configuring Virtual Networks in Cloud

- Set up a **Virtual Private Cloud (VPC) / Azure Virtual Network / GCP VPC**
  Configure **subnets, firewalls, and security groups**
  Test **network communication between two VMs**

Lab Assignment 20: Deploying a Web Application on Cloud

Deploy a Python/Node.js/Java web app using:

- **AWS Elastic Beanstalk**
- **Azure App Service**
- **Google App Engine**
  Connect the app to Cloud Database (RDS, CosmosDB, Firestore)
  Monitor application performance and logs

Targeted Application & Tools that can be used:
Text Book(s): 1. John Rittinghouse and James Ransome, “Cloud Computing, Implementation, Management and Security”, CRC Press. 2. Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi, “Mastering Cloud Computing”, McGraw Hill Education.
Reference(s): 1. David E.Y. Sarna, “Implementing and Developing Cloud Applications”, CRC Press. 2. Anthony T Velte, Toby J Velte, Robert Elsenpeter, “Cloud Computing: A Practical Approach”, Tata McGraw-Hill. Web resources: https://presiuniv.knimbus.com/user#/home

Course Code: CSE1505	Course Title: Web Technologies Laboratory Type of Course: Program core lab course	L-T-P- C	0	0	2	1
Version No.	1.0					
Course Pre-requisites	Database Management Systems-CSE3156					
Anti-requisites	**NIL**					
Course Description	This course highlights the comprehensive introduction to scripting languages that are used for creating web-based applications. The associated laboratory provides an opportunity to implement the concepts and enhance critical thinking and analytical skills.					
Course Objective	**The objective of the course is to familiarize the learners with the concepts of Web Technology and attain Skill Development through Experiential Learning techniques.**					
Course Outcomes	**On successful completion of this course the students shall be able to:** **CO1:** Implement web-based application using client-side scripting languages. **(Apply)** **CO2**: Apply various constructs to enhance the appearance of a website. **(Apply)** **CO3:** Apply server-side scripting languages to develop a web page linked to a database. **(Apply)**					
Course Content:						

List of Laboratory Tasks:

Experiment No. 1: Demonstration of XHTML features

Level 1: Demonstration of various XHTML Tags (Level 1)

Level 2: Design and develop static web pages for an online Book store (Level 2).

Experiment No. 2: Application of CSS in web designing

Level 1: Design a document using XHTML and CSS to create a catalog of items for online electronic shopping.

Level 2: Create and save XML document for students' information and display the same using cascaded style sheet.

Experiment No. 3: Application of PHP in web designing.

Level 1: Write a PHP program to read the personal information of a person such as first name, last name, age, permanent address, and pin code entered by the user into a table created in MySQL. Read the same information from the database and display it on the front end.

Level 2: Using PHP develop a web page that accepts book information such as ISBN number, title, authors, edition, and publisher and store information submitted through the web page in MySQL database.

Experiment No. 4: Building a website.

Build a website for organizing an International Conference. The conference website must be able to collect the author's details and upload a file.

Targeted Application & Tools that can be used: Xampp web server to be used to demonstrate PHP.

Project work/Assignment:

Assignments are given after completion of each module which the student need to submit within the stipulated deadline.

Textbook(s):

1] Robert. W. Sebesta, "*Programming the World Wide Web"*, Pearson Education, 9th Edition, 2016.

2]Paul Deitel, Harvey Deitel, Abbey Deital,"Internet & World Wide Web How to Program", Fifth Edition, Pearson Education, 2021.

3]CSS Notes for Professionals, ebook available at https://books.goalkicker.com/CSSBook/ (Retrieved on Jan. 20, 2022)

4]Deitel, Deitel, Goldberg,"*Internet & World Wide Web How to Program*", Fifth Edition, Pearson Education, 2021.

Reference Book(s):

R1. Randy Connolly, Ricardo Hoar,"Fundamentals of Web Development", Pearson Education India, 1st. Edition.2016.

R2. Jeffrey C. Jackson,"Web Technologies: A Computer Science Perspective", Pearson Education, 1st Edition,2016.

Additional web-based resources

W1. W3schools.com

W2. Developer.mozilla.org/en-US/docs/Learn

W3. docs.microsoft.com

W4. informit.com/articles/ The Relationship Between Web 2.0 and Social Networking

https://presiuniv.knimbus.com/user#/home

Topics related to development of "FOUNDATION":

Web, WWW, Web browsers, Web servers, Internet.

CSS, PHP.

Designing the website for healthcare.

The objective of the course is to familiarize the learners with the concepts of Web Technology and attain Skill Development through Experiential Learning techniques.

Course Code:CSE 7100	**Course Title: Mini Project** **Type of Course:**	**L- T-P- C**	0	0	0	4
Version No.	**1.0**					
Course Pre-requisites	**Knowledge and Skills related to all the courses studied in previous semesters.**					
Anti-requisites	**NIL**					
Course Description	Students observe science and technology in action, develop an awareness of the method of scientific experimentation, and often get an opportunity to see, study and operate sophisticated and costly equipment. They also learn about the implementation of the principles of management they have learnt in class, when they observe multidisciplinary teams of experts from engineering, science, economics, operations research, and management deal with techno-economic problems at the micro and macro levels. Finally, it enables them to develop and refine their language, communication and inter-personal skills, both by its very nature, and by the various evaluation components, such as seminar, group discussion, project report preparation, etc. The broad-based core education, strong in mathematics and science and rich in analytical tools, provides the foundation necessary for the student to understand properly the nature of real-life problems. The students have options to pursue this course as either Project Work and Dissertation at the university, or Project Work in an Industry/ Company/ Research Laboratory, or Internship Program in an Industry/Company.					
Course Objectives	The objective of the course is to familiarize the learners with the concepts of Professional Practice and attain Employability Skills through Experiential Learning techniques.					
Course Outcomes	On successful completion of this course the students shall be able to: 1. Identify the engineering problems related to local, regional, national or global needs. (Understand) 2. Apply appropriate techniques or modern tools for solving the intended problem. (Apply) 3. Design the experiments as per the standards and specifications. (Analyze) 4. Interpret the events and results for meaningful conclusions. (Evaluate) 5. Appraise project findings and communicate effectively through scholarly publications. (Create)					
Catalogue prepared by	Dr. Sampath A K					
Recommended by the Board of Studies on						

Date of Approval bythe Academic Council	

Course Code:CSE 7300	**Course Title:** Capstone Project **Type of Course:**	**L- T-P- C**	0	0	0	10
Version No.	**1.0**					
Course Pre-requisites	**Knowledge and Skills related to all the courses studied in previous semesters.**					
Anti-requisites	**NIL**					
Course Description	Students observe science and technology in action, develop an awareness of the method of scientific experimentation, and often get an opportunity to see, study and operate sophisticated and costly equipment. They also learn about the implementation of the principles of management they have learnt in class, when they observe multidisciplinary teams of experts from engineering, science, economics, operations research, and management deal with techno-economic problems at the micro and macro levels. Finally, it enables them to develop and refine their language, communication and inter-personal skills, both by its very nature, and by the various evaluation components, such as seminar, group discussion, project report preparation, etc. The broad-based core education, strong in mathematics and science and rich in analytical tools, provides the foundation necessary for the student to understand properly the nature of real-life problems. The students have options to pursue this course as either Project Work and Dissertation at the university, or Project Work in an Industry/ Company/ Research Laboratory, or Internship Program in an Industry/Company.					
Course Objectives	The objective of the course is to familiarize the learners with the concepts of Professional Practice and attain Employability Skills through Experiential Learning techniques.					
Course Outcomes	On successful completion of this course the students shall be able to: 1. Identify problems based on societal /research needs. (Understand) 2. Apply Knowledge and skill to solve societal problems in a group. (Apply) 3. Develop interpersonal skills to work as member of a group or leader. (Apply) 4. Analyze the inferences from available results through theoretical / Experimental / Simulations. (Analyze) 5. Analyze the impact of solutions in societal and environmental context for sustainable development. (Analyze) 6. Improve in written and oral communication. (Create) 7. Demonstrate capabilities of self-learning in a group, which leads to lifelong learning. (Understand)					
Catalogue prepared by	Dr. Sampath A K					

Recommended by the Board of Studies on	
Date of Approval bythe Academic Council	

Course Code: CSE3400	Course Title: Intelligent Systems with Machine Learning Type of Course: 1] Program Core 2] Laboratory integrated	L- T-P-C	2	0	2	3

Version No.		1.0
Course Pre-requisites		CSE1700 – Essentials of AI
Anti-requisites		**NIL**
Course Description		Machine Learning algorithms are the key to develop intelligent systems such as Apple's Siri, Google's self-driving cars etc. This course introduces the concepts of the core machine learning techniques such as Regression learning, Bayesian learning, Ensemble learning, Perceptron learning, Unsupervised learning, Competitive learning, learning from Gaussian mixture models and learning to detect outliers. Course lectures covers both the theoretical foundations as well as the essential algorithms for the various learning methods. Lab sessions complement the lectures and enable the students in developing intelligent systems for real life problems.
Course Objectives		This course is designed to improve the learners 'EMPLOYABILITY SKILLS' by using EXPERIENTIAL LEARNING techniques. The supervised hands-on laboratory exercises, assessments and the group projects facilitate this learning process.
Course Out Comes		On successful completion of the course the students shall be able to: 1] Apply advanced supervised machine learning methods for predictive modeling. [Application] 2] Produce machine learning models with better predictive performance using meta learning algorithms [Application] 3] Create predictive models using Perceptron learning algorithms[Application] 4] Employ advanced unsupervised learning algorithms for clustering, competitive learning and outlier detection[Application] 5] Implement machine learning based intelligent models using Python libraries. [Application]
Course Content:		

Module 1	Supervised Learning	Assignment		Programming using Keras/Sklearn	**No. of Classes L – 7 P – 12**
	Topics: An overview of Machine Learning(ML); ML workflow; types of ML; Types of features, Feature Engineering -Data Imputation Methods; Regression – introduction; simple linear regression, loss functions; Polynomial Regression; Logistic Regression; Softmax Regression with cross entropy as cost function**; Bayesian Learning** – Bayes Theorem, estimating conditional probabilities for categorical and continuous features, Naïve Bayes for supervised learning; Bayesian Belief networks; Support Vector Machines – soft margin and kernel tricks.				
Module 2	Ensemble Learning	Assignment		Programming using	**No.**

				Keras/Sklearn	**of Classes L-3 P-4**
	Topics: **Ensemble Learning** – using subset of instances – Bagging, Pasting, using subset of features –random patches and random subspaces method; Voting Classifier, Random Forest; Boosting – AdaBoost, Gradient Boosting, Extremely Randomized Trees, Stacking.				
Module 3	Perceptron Learning	Assignment /Quiz		Programming using Keras/Sklearn	**No. of Classes L-7 P -2**
	Topics: **Perceptron Learning** – from biological to artificial neurons, Perceptrons, Linear Threshold Units, logical computations with Perceptrons, common activation functions – sigmoid, tanh, relu and softmax, common loss functions, multi-layer Perceptrons and the Backpropagation algorithm using Gradient Descent.				
Module 4	Unsupervised Learning	Assignment		Programming using Keras/Sklearn	**No. of Classes L-6 P -6**
	Topics: **Unsupervised Learning** – simple k Means clustering- simple and mini-batch**;** updating centroids incrementally; finding the optimal number of clusters using Elbow method ; Silhoutte coefficient,drawbacks of kMeans,kMeans++ **;** Divisive hierarchical clustering – bisecting k-means, clustering using Minimum Spanning Tree (MST) **Competitive Learning** - Clustering using Kohenen's Self Organising Maps (SOM), **Density Based Spatial Clustering – DBSCAN**; clustering using Gaussian Mixture Models (GMM) with EM algorithm ; Outlier Detection methods – **Isolation Forest, Local Outlier Factor(LOF)**				
	List of Laboratory Tasks: **Experiment N0 1: Methods for handling missing values** **Level 1:** Given a data set from UCI repository, implement the different ways of handling missing values in it using Scikit-learn library of Python **Level 2:** Implement one of these methods using a custom defined function in Python. **Experiment No. 2: Data Visualization** **Level 1** Perform Exploratory Data Analysis for a given data set by creating Scatter Plot, Pair Plot, Count Plot using Matplotlib and Seaborn **Level 2** Create Heat Maps, WordCloud **Experiment No. 3: Regression learning** **Level 1** Given a data set from UCI repository, implement the simple linear regression algorithm and estimate the models parameters and the performance metrics. Plot the learning curves. **Level 2** Implement the polynomial regression algorithm. Compare the learning curves of Polynomial and Linear Regression. **Experiment No.4: Logistic regression** **Level 1** Write custom code for generating the logistic/sigmoid plot for a given input **Level 2** Given a data set from UCI repository, implement the Logistic regression algorithm. Estimate the class probabilities for a given test data set. Plot and analyze the decision boundaries. **Experiment No.5: Bayesian Learning** **Level 1** Given a data set from UCI repository, implement a classification model using the Bayesian algorithm **Experiment No.6: Support Vector Machine(SVM)** **Level 1** Given data sets from UCI repository, implement a linear SVM and a non-linear SVM based				

	classification model. **Experiment No. 7: Ensemble Learning** **Level 1** : Implement Ensemble Learning algorithms such as Bagging, Pasting and Out-of Bag Evaluation **Level 2** : Random Patches and Random Subspace Method **Experiment No. 8: Ensemble Learning** **Level 1** : AdaBoost and Gradient Boosting, Stacking **Experiment No. 9: Perceptron Learning** **Level 1** : Implement the Perceptron Classifier **Level 2** : – An Image Classifier Using the Sequential API of Keras **Experiment No. 10: Unsupervised Learning** **Level 1** : K-means – simple and mini-batch. Finding the optimal number of clusters using Elbow method and Silhoutte Coefficient . Compare the inertia of both as k increases. Tuning the hyperparameter 'k' using GridSearchCV. **Level 2** : – Using clustering for Image segmentation and Preprocessing. Kmeans++ **Experiment No. 11: Density Based Clustering** **Level 1** Implement DBSCAN – clustering using the local density estimation. Perform hard and soft clustering for new instances. **Experiment No. 12: Outlier Detection** **Level 1** Outlier Detection using Isolation Forest and Local Outlier Factor
	Targeted Application & Tools that can be used : 1. Execution of the ML algorithms will be done using the Google's cloud service namely "Colab", available at https://colab.research.google.com/ or Jupyter Notebook. 2. The data sets will be from the bench marking repositories such as UCI machine learning repository available at : https://archive.ics.uci.edu/ml/index.php 3. Laboratory tasks will be implemented using the libraries available in Python such as Scikit learn, matplotlib, seaborn, perceptron and the deep learning framework namely Keras.
	Project work/Assignment: Mention the Type of Project /Assignment proposed for this course
	Students can be assigned a mini project to develop a machine learning application for real-life problems in various domains such as health care, business intelligence, environmental modeling, etc.
	Text Book There are a number of useful textbooks for the course, but each cover only a part of the course syllabus. Following is an indicative list of textbooks. 1. Aurélien Géron, "Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow", Oreilly, Second Edition, 2019. 2. Andreas C Muller, Sarah Guido, "Introduction to Machine Learning with Python :A Guide for Data Scientists", Oreilly, First Edition, 2018 3. Giuseppe Bonaccorso, "Machine Learning Algorithms: A reference guide to popular algorithms from data science and machine learning", Packt Publishing, 2017.
	References In references apart from the books and web links, mention a few standards &Hand books relevant to the Laboratory tasks used by the professionals. 1. Tan P. N., Steinbach M & Kumar V. "***Introduction to Data Mining"*, Pearson Education, 2016. 2. **https://towardsdatascience.com/machine-learning/home**

	3. MITopencourseware:https://ocw.mit.edu/courses/6-0002-introduction-to-computational-thinking-and-data-science-fall-2016/resources/lecture-11-introduction-to-machine-learning/ 4. https://onlinecourses.nptel.ac.in/noc21_cs85/preview

urse Code: CSE3401	**Course Title: Advanced Deep Learning Techniques** **e of Course: Theory & Integrated Laboratory**	**-P- C**	**3**	**0**	**0**	**3**
sion No.						
urse Pre-requisites	CSE1700 – Essentials of AI					
i-requisites						
urse Description	This course introduces students to the concepts of deep neural networks and state of the art approaches to develop deep learning models. In this course students will be given an exposure to the details of neural networks as well as deep learning architectures and to develop end-to-end models for such tasks. It will help to design and develop an application-specific deep learning models and also provide the practical knowledge handling and analyzing end user realistic applications. Topics include Fundamental concepts of deep neural networks, Convolutional Neural Networks, Recurrent Network structures, Deep Unsupervised Learning, Generative Adversarial Networks and applications in various problem domains.					
urse Objective	This course is designed to improve the learners EMPLOYABILITY SKILLS by using EXPERIENTIAL LEARNING techniques.					
urse Outcomes	On successful completion of this course the students shall be able to: 1. Learn the Fundamental Principles of Deep Learning. (Remember). 2. Identify the Deep Learning Algorithms for learning tasks in various related domains (Apply). 3. To understand and apply deep generative models. (Understand). 4. Apply deep learning architectures to image and audio data. (Apply)					
Course Content:						
dule 1	**Introduction to Deep Learning and Neural Networks**	ignment			**13[7L+6P] Sessions**	
ics: Fundamentals of Deep Learning, Perceptron, Multilayer Perceptron, Optimizing Perceptions using Activation Functions, Loss Functions, Gradient Descent. Feedforward Neural Network, Training Neural Network with Back-propagation, Hyper parameters, Regularization, Dropouts, Batch Normalization, Practical Issues in Neural Network Training -The Problem of Overfitting, The Vanishing and Exploding Gradient Problems						

dule 2	**Common Deep Learning Architectures:**	ignment		**18[8L+10P] Sessions**

ics:

Convolutional Neural Network, Transfer learning Techniques, Variants of CNN: DenseNet, ResNet

Sequence Modelling: Recurrent Neural Network and its variants - Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU)

dule 3	**Deep Generative Models**	ignment		**16[8L+8P] Sessions**

Topics:
Generative Adversarial Networks, Kohonen Networks, Autoencoders, Boltzmann Machine, Restricted Boltzmann Machine, Deep Belief Network

dule-4	**Advanced Deep Learning Architectures**	ignment		**13[7L+6P] Sessions**

Topics:
Hopfield Network, Probabilistic Neural Network, Deep Reinforcement Learning - The Basic Framework of Reinforcement Learning

Deep Learning applications: Image segmentation, Object detection,Speech Recognition, Video Analytics

ject work/Assignment:

1. **Assignment 1 on (Module 1 and Module 2)**
2. **Assignment 2 on (Module 3 and Module 4)**

List of Laboratory Tasks:

Lab 1: Working with Deep Learning Frameworks
Objective: Explore various Deep Learning Frameworks
Tasks: Identify deep learning frameworks (Keras, Tensorflow, Matplotlib, etc)
Activity: Practice with various methods available in DL Frameworks to develop a Model.

Lab 2: Build a Basic Artificial Neural Network
Objective: Create a ANN with DL frameworks.
Task: Identify suitable ANN Layers using Keras and Tensorflow.
Activity: Design a basic Artificial Neural Networks using Keras with TensorFlow (pima-indians-diabetes)

Lab 3 and Lab 4: Build a MultiLayer Perceptron
Objective: Create a MLP for classification task.
Task: Identify suitable model for house price prediction.
Activity: Design a MLP for implementing classification and fine-tuning using House price.csv

Lab 5: Build a Convolutional Neural Network
Objective: Create a CNN model.
Task: Build CNN architecture for Dog-Cat classification problem.
Activity: Implement a Convolution Neural Network (CNN) for dog/cat classification problem using keras

Lab 6 and Lab 7: Build a Time-Series Model
Objective: Create a RNN and LSTM Model

Task: Build RNN/LSTM Model for predicting time series data.
Activity Train a sentiment analysis model on IMDB dataset, use RNN layers with LSTM/GRU notes

Lab 8: Build a Gated Recurrent Unit architecture.
Objective: Create a Time Series Model.
Task: Build GRU Architecture for predicting time series data.
Activity: Implement a GRU architecture for language translations.

Lab 9 and Lab 10: Build a Transfer Learning Model.
Objective: Create a Seq2Seq Model
Task: Create Hugging-face API using Transfer learning model.
Activity: Implement Transfer Learning models for classification problems Exploring Hugging-face API

Lab 11: Build an Auto-Encoder model
Objective: Create an Unsupervised Deep Learning Model.
Task: Create AutoEncoder network Output Translations.
Activity: implement an Encoder-Decoder Recurrent neural network model for Neural Machine Translation.

Lab 12: Build Generative Adversarial Networks.
Objective: Create an Unsupervised Deep Learning Model.
Task: Design GAN Architecture for Image generations.
Activity: Design a Age Prediction model by Applying Generative Adversarial

REFERENCE MATERIALS:

TEXTBOOKS

1. François Chollet, “Deep Learning with Python”, 2nd Edition, Manning Publications, 2022
2. Ian Goodfellow, Yoshua Bengio, Aaron Courville, “Deep Learning”, MIT Press, 2017.

REFERENCES

1. Amlan Chakrabarti Amit Kumar Das, Saptarsi Goswami, Pabitra Mitra , “Deep Learning”, Pearson Publication, 2021.
2. David Foster, “Generative Deep Learning” O’Reilly Publishers, 2020.
3. John D Kellehar, “Deep Learning”, MIT Press, 2020.

JOURNALS/MAGAZINES

1. IEEE Transactions on Neural Networks and Learning Systems
https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=5962385
2. IEEE Transactions on Pattern Analysis and Machine Intelligence
https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=34http://ijaerd.com/papers/special_papers/IT032.pdf
3. International Journal of Intelligent Systems https://onlinelibrary.wiley.com/journal/1098111x

SWAYAM/NPTEL/MOOCs:

4. Swayam Nptel – Deep Learning – IIT Ropar https://onlinecourses.nptel.ac.in/noc21_cs35/preview
5. Coursera – Neural Networks and Deep Learning Andrew Ng
6. Coursera - Neural Networks for Machine Learning by Geoffrey Hinton in Coursera

Course Code:CSE3402	**rse Title: Computational Optimization for Intelligent Systems** **e of Course: Discipline Elective**	**-- P- C**	3	0	)0	3

sion No.		
rse Pre-requisites		CSE1700 – Essentials of AI
i-requisites		
rse Description		This course introduces a range of machine learning models and optimization tools that are used to apply these models in practice. Course will introduce what lies behind the optimization tools often used as a black box as well as an understanding of the trade-offs of numerical accuracy and theoretical and empirical complexity. For the students with some optimization background this course will introduce a variety of applications arising in machine learning and statistics as well as novel optimization methods targeting these applications.
rse Objective		The objective of the course is to familiarize the learners with the concepts of **Optimization Techniques for Machine Learning** and attain **Skill Development** through **Participative Learning** techniques.
rse Outcomes		On successful completion of this course the students shall be able to: 1. **Demonstrate** simple examples to illustrate how Machine Learning is applied in real-world scenarios. [**Understand].** 2. **Implement** Machine Learning models (e.g., decision trees, linear regression, neural networks) using tools or programming languages. **[Apply].** 3. **Determine** the suitability of convex optimization in solving problems like portfolio optimization, machine learning, or network design. **[Apply].** 4. **Solve** convex optimization problems with real or simulated data, such as minimizing a cost function or optimizing resource allocation. **[Apply].**
rse Content:		

Module 1:	**damentals of Machine learning**	**z**		wledge based Quiz	**0 Sessions**
	Topics: Machine Learning Paradigm, Empirical Risk Minimization, Structural Risk Minimization, Learning Guarantees, Introduction of VC-Dimension, Dimensionality Reduction Techniques.				
dule 2:	**chine learning models**	**z**		nprehension based Quiz	**2 Sessions**
	Topics: Logistic Regression, Support Vector Machines, Sparse Regression, Low Dimensional Embedding, Low Rank Matrix Factorization, Sparse PCA, Multiple Kernel Learning, Loss Functions, Entropy, Cross-Entropy Loss				

dule 3	**vex optimization models**	**ignment**		ch-wise Assignments	**3 Sessions**
	Topics: Linear Optimization, Convex Quadratic Optimization, Second Order Cone Optimization, Semi-definite Optimization, Convex Composite Optimization.				
dule 4:	**thods for convex optimization**	ignment and Presentation		ch-wise Assignment and Presentations	**0 Sessions**
	Topics: gradient descent, Newton method, interior point methods, active set, prox methods, accelerated gradient methods, coordinate descent, cutting plances, stochastic gradient.				
	geted Application & Tools that can be used: Use of Google Colab				
	ject work/Assignment: **vey on Methods for convex optimization** **vey on Machine learning models related to optimization**				
	t Book T1. Charu C. Aggarwal, "*Linear Algebra and Optimization for Machine Learning*", Springer, 2020. T2. Sra Suvrit, Nowozin Sebastian, and Wright Stephen J, "*Optimization for Machine Learning*", The MIT Press,2012.				
	erences R1.Guanghui Lan, "*First-order and Stochastic Optimization Methods for Machine Learning*", Springer Cham, 2020. **Web References** W1. https://sm-nitk.vlabs.ac.in/ W2. https://nptel.ac.in/courses/				
	Topics relevant to SKILL DEVELOPMENT: Concepts of Convex optimization models and Methods for convex optimization f**or Skill Development through Problem Solving methodologies. This is attained through assessment component mentioned in course handout.**				
Catalogue prepared by		Dr.Nagaraja S R			

Course Code: CSE3403	**Course Title: Reinforcement Learning for AI Systems** **Type of Course: Theory Only**	**L-T-P-C**	2	0	2	3
Version No.	1.0					
Course Pre-requisites	• CSE1700 – Essentials of AI					
Anti-requisites	NIL					

Course Description		The goal of this class is to provide an introduction to reinforcement learning, a very active research sub-field of machine learning. Reinforcement learning is concerned with building programs that learn how to predict and act in a stochastic environment, based on past experience. Applications of reinforcement learning range from classical control problems, such as power plant optimization or dynamical system control, to game playing, inventory control, and many other fields. Notably, reinforcement learning has also produced very compelling models of animal and human learning. During this course, we will study theoretical properties and practical applications of reinforcement learning. We will follow the second edition of the classic textbook by Sutton & Barto (available online for free, or from MIT Press), and supplement it as needed with papers and other materials.
Course Objective		The objective of the course is to familiarize the learners with the concepts of **Reinforcement Learning** and attain **Skill Development** through **Problem Solving** Methodologies.
Course Out Comes		On successful completion of the course the students shall be able to: 1. Knowledge of basic and advanced reinforcement learning techniques. 2. Identification of suitable learning tasks to which these learning techniques can be applied. 3. Appreciation of some of the current limitations of reinforcement learning techniques. 4. Formulation of decision problems, set up and run computational experiments, evaluation of results from experiments.
Course Content:		

Module 1	Introduction	Assignment		Programming	**No. of Classes:10**
	Topics: Course logistics and overview. Origin and history of Reinforcement Learning research. Its connections with other related fields and with different branches of machine learning. **Probability Primer** Brush up of Probability concepts - Axioms of probability, concepts of random variables, PMF, PDFs, CDFs, Expectation. Concepts of joint and multiple random variables, joint, conditional and marginal distributions. Correlation and independence.				
Module 2	Markov Decision Process	Assignment		Programming	**No. of Classes:10**
	Topics: Introduction to RL terminology, Markov property, Markov chains, Markov reward process (MRP). Introduction to and proof of Bellman equations for MRPs along with proof of existence of solution to Bellman equations in MRP. Introduction to Markov decision process (MDP), state and action value functions, Bellman expectation equations, optimality of value functions and policies, Bellman optimality equations.				
Module 3	Prediction and Control by Dynamic Programing	Assignment		Programming	**No. of Classes:10**
	Topics: Overview of dynamic programing for MDP, definition and formulation of planning in MDPs, principle of optimality, iterative policy evaluation, policy iteration, value iteration, Banach fixed point theorem, proof of contraction mapping property of Bellman expectation and optimality operators, proof of convergence of policy evaluation and value iteration algorithms, DP extensions **Monte Carlo Methods for Model Free Prediction and Control** Overview of Monte Carlo methods for model free RL, First visit and every visit Monte Carlo, Monte Carlo control, On policy and off policy learning, Importance sampling.				
Module 4	TD Methods and Policy Gradients	Assignment		Programming	**No. of Classes:10**

	Topics: Incremental Monte Carlo Methods for Model Free Prediction, Overview TD(0), TD(1) and TD(λ), k-step estimators, unified view of DP, MC and TD evaluation methods, TD Control methods - SARSA, Q-Learning and their variants. Getting started with policy gradient methods, Log-derivative trick, Naive REINFORCE algorithm, bias and variance in Reinforcement Learning, Reducing variance in policy gradient estimates, baselines, advantage function, actor-critic methods.
	Targeted Application & Tools that can be used: While Convolution Neural Network (CNN) and Recurrent Neural Network (RNN) are becoming more important for businesses due to their applications in Computer Vision (CV) and Natural Language Processing (NLP), Reinforcement Learning (RL) as a framework for computational neuroscience to model decision making process seems to be undervalued. Besides, there seems to be very little resources detailing how RL is applied in different industries. Despite the criticisms about RL's weaknesses, RL should never be neglected in the space of corporate research given its huge potentials in assisting decision making. **Tools**: Torch, Google Colaboratory, Spider, Jupiter Notebook
	Project work/Assignment:
	This part is written for general readers. At the same time, it will be of greater value for readers with some knowledge about RL. • **Resources management in computer clusters** Designing algorithms to allocate limited resources to different tasks is challenging and requires human-generated heuristics. The paper "Resource Management with Deep Reinforcement Learning" [2] showed how to use RL to automatically learn to allocate and schedule computer resources to waiting jobs, with the objective to minimize the average job slowdown. State space was formulated as the current resources allocation and the resources profile of jobs. For action space, they used a trick to allow the agent to choose more than one action at each time step. Reward was the sum of (-1/duration of the job) over all the jobs in the system. Then they combined REINFORCE algorithm and baseline value to calculate the policy gradients and find the best policy parameters that give the probability distribution of actions to minimize the objective. • **Traffic Light Control** Researchers tried to design a traffic light controller to solve the congestion problem. Tested only on simulated environment though, their methods showed superior results than traditional methods and shed a light on the potential uses of multi-agent RL in designing traffic system. Five agents were put in the five-intersection traffic network, with a RL agent at the central intersection to control traffic signalling. The state was defined as eight-dimensional vector with each element representing the relative traffic flow of each lane. Eight choices were available to the agent, each representing a phase combination, and the reward function was defined as reduction in delay compared with previous time step. The authors used DQN to learn the Q value of the {state, action} pairs. • **Robotics** There are tremendous works on applying RL in Robotics. Readers are referred to for a survey of RL in Robotics. In particular, trained a robot to learn policies to map raw video images to robot's actions. The RGB images were fed to a CNN and outputs were the motor torques. The RL component was the guided policy search to generate training data that came from its own state distribution. • **Web System Configuration** There are more than 100 configurable parameters in a web system and the process of tuning the parameters requires a skilled operator and numerous trail-and-error tests. The paper "A Reinforcement Learning Approach to Online Web System Auto-configuration" showed the first attempt in the domain on how to do autonomic reconfiguration of parameters in multi-tier web systems in VM-based dynamic environments. The reconfiguration process can be formulated as a finite MDP. The state space was the system configuration, action space was {increase, decrease, keep} for each parameter, and reward was defined as the difference between the given targeted response time and measured response time. The authors used the model-free Q-learning algorithm to do the task.

	Text Book 1. "Reinforcement Learning: An Introduction", Richard S. Sutton and Andrew G. Barto, 2nd Edition 2. "Probability, Statistics, and Random Processes for Electrical Engineering", 3rd Edition, Alberto Leon-Garcia 3. "Machine Learning: A Probabilistic Perspective", Kevin P. Murphy
	References 1. Richard S. Sutton and Andrew G. Barto, "Reinforcement learning: An introduction", Second Edition, MIT Press, 2019. 2. Li, Yuxi. "Deep reinforcement learning." arXiv preprint arXiv:1810.06339 (2018). 3. Wiering, Marco, and Martijn Van Otterlo. "Reinforcement learning." Adaptation, learning, and optimization 12 (2012):
	E-Resources NPTEL course – https://onlinecourses.nptel.ac.in/noc19_cs55/preview https://archive.nptel.ac.in/courses/106/106/106106143/ https://www.digimat.in/nptel/courses/video/106106143/L35.html
	Topics relevant to "SKILL DEVELOPMENT": Real time Data Analysis using Reinforcement learning **for Skill Development through Problem Solving techniques. This is attained through assessment component mentioned in course handout.**

Course Code: CSE3404	**Course Title: Computational Linguistics and Natural Language Processing** **Type of Course: Theory Only Course**	**L- T-P- C**	3	0	0	3
Version No.	1.0					
Course Pre-requisites	CSE1700 – Essentials of AI					
Anti-requisites	NIL					
Course Description	The purpose of this course is to introduce students to the science of natural language processing (NLP). NLP is the science of extracting information from unstructured text. It is basically how we can teach machines to understand human languages and extract meaning from text. In addition to regular theory, the course also involves: 1. Programming Assignments 2. Regular Quiz Tests (once a week and once after every module)					

Course Objective		The objective of the course is to familiarize the learners with the concepts of Fundamentals of Natural language Processing and attain **Skill Development** through **Participative Learning** techniques.				
Course Out Comes		**On successful completion of the course the students shall be able to:** • **Understand** the fundamental concepts of Natural Language Processing. [Knowledge] • **Read** corpora and **train** models for different NLP tasks. [Application] • **Use** word embeddings for solving an NLP Application. [Application] • **Understand** sequence to sequence modeling as used in machine translation. [Application]				
Course Content:						
Module 1		**Introduction**	Quizzes			**7 Sessions**
	Topics: Introduction. History. Text Analytics. Various tasks in NLP. Sentence boundary Detection. Edit distance. Introduction to word embeddings, PoS tagging, chunking, parsing, machine translation.					
Module 2		**Word and Text Representations**	Quizzes		Assignments	**8 Sessions**
	Topics: Logistic Regression and Naïve Bayes classification. Vector semantics and embeddings. Neural Networks and Neural Language Models. Text representations and classification. Deep learning architectures for sequence processing (CNN and LSTM).					
Module 3		**PoS Tagging, NER Tagging and Parsing**	Quizzes		Assignments	**12 Sessions**
	Topics: Part-of-Speech Tagging – using NLTK and spacy. Building a PoS Tagger using existing data and Hidden Markov Model. Named Entity Recognition. Relationship between NER tagging and PoS tagging. Constituency Parsing.					
Module 4		**NLP Applications**	Quizzes			**9 Sessions**
	Topics: Lexical Resource Creation. Sentiment Analysis. Machine Translation. Word Sense Disambiguation and WordNet. Question Answering.					
	Targeted Application & Tools that can be used: 1. **Python Libraries (Eg. NLTK, Spacy, etc.)** 2. **Java (Stanford CoreNLP)** 3. **Google Colab**					
	Project work/Assignment:					
	Assignment: Students will have to do group assignments for Modules 2 & 3. As a part of their assignments, they will have to implement the solution to particular problems.					
	Text Book **T1**Daniel Jurafsky, and James Martin."*Speech and Language Processing*" (3rd edition draft, 2022)					
	References **1**Chris Manning and HinrichSchutze, "*Foundations of Statistical Natural Language Processing*", 1st Edition, MIT Press. 1999. **2**PawanGoyal, "*Natural Language Processing*". NPTEL.					

	E-Book Link for R2: https://drive.google.com/file/d/10nbwAJd-dv6htOOZVBgAvLd1WscI0RqC/view **Web resources:https://web.stanford.edu/~jurafsky/slp3/** **NPTEL Course: https://onlinecourses.nptel.ac.in/noc22_cs98/course**
	Topics relevant to "SKILL DEVELOPMENT": Assignment implementations in software, batch wise presentations for developing Skill Development through Participative Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE3405	**Course Title: Synergistic Neural Fuzzy Computing** **Type of Course: Discipline Elective in AI & ML Basket** **Theory Course**		**L-T-P-C**	2	0	2	3
Version No.	1.0						
Course Pre-requisites	CSE1700 – Essentials of AI						
Anti-requisites	NIL						
Course Description	This course aims to introduce the basic concepts of Neural Networks and Fuzzy Logic. Neural networks reflect the behavior of the human brain, allowing computer programs to recognize patterns and solve common problems in the fields of AI, machine learning, and deep learning. Fuzzy Logic is a method of reasoning that resembles human reasoning. The approach of Fuzzy Logic imitates the way of decision-making in humans that involves all intermediate possibilities between digital values YES and NO. This course introduces fundamental concepts in Neural Networks and Fuzzy Logic Theory.						
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Neural Networks and Fuzzy Logic** and attain **Skill Development** through **Participative Learning** techniques.						
Course Outcomes	**On successful completion of this course the students shall be able to:** 1. Define the concept of Neural Networks. [Knowledge] 2. Define the ideas behind most common learning algorithms in Neural Network.[Knowledge] 3. Discuss the concepts of Fuzzy Sets and Relations. [Comprehension] 4. Demonstrate the Fuzzy logic concepts and its applications.[Application]						
Course Content:							
Module 1	**Introduction to Neural Network**	Quiz		Single Layer Perceptron		**9Classes**	
	Topics: Introduction to NN: History, Artificial and biological neural networks, Artificial intelligence and neural networks. Neurons and Neural Networks: Biological neurons, Models of single neurons, Different neural network models. Single Layer Perceptron: Least mean square algorithm, Learning curves, Learning rates, Perceptron.						
Module 2	**Multilayer Perceptron**	Quiz		Multilayer Perceptron		**10 Classes**	
	Topics: Multilayer Perceptron: The XOR problem, Back-propagation algorithm, Heuristic for improving the back-						

	propagation algorithm, Some examples. Radial-Basis Function Networks: Interpolation, Regularization, Learning strategies. Kohonen Self-Organising Maps: Self-organizing map, The SOM algorithm, Learning vector quantization.					
Module 3	**Fuzzy Sets, Operations and Relations**	Quiz		Fuzzy Operations	**10Classes**	
	Topics: Fuzzy Sets: Crisp Sets - an Overview, Fuzzy Sets - Definition and Examples, α - Cuts and its Properties, Representations of Fuzzy Sets, Extension Principles of Fuzzy Sets. Fuzzy Operations: Operations on Fuzzy Sets - Fuzzy Complements, Fuzzy Intersections, Fuzzy Unions, Combinations of Operations, Aggregation Operations. Fuzzy Relations: Binary Fuzzy relations, Fuzzy Equivalence Relations, Fuzzy Compatibility Relations.					
Module 4	**Fuzzy Logic and Fuzzy Logic Controller**	Assignment		Developing Fuzzy Logic Controller	**10Classes**	
	Fuzzy Logic: Classical Logic, Multivalued Logic, Fuzzy Propositions, Fuzzy Quantifiers, Linguistic Hedges, Inference from Conditional Fuzzy Propositions, Conditional and Qualified Propositions and Quantified Propositions. Fuzzy Controllers: An Overview, Fuzzification Module, Fuzzy Rule Base, Fuzzy Inference Engine, Defuzzification Module, An Example.					
	Targeted Application & Tools that can be used: 1. Python Libraries and Software (Eg.,Tensorflow, Scikit-Learn etc.) 2. Matlab (Neural Network Toolbox, Fuzzy Logic Toolbox)					
	Project work/Assignment:					
	Students will have to do group assignments for Modules 2 & 4. As a part of their assignments, they will have to implement the solution to particular problems.					
	Textbook(s): 1. Haykin, Simon. "*Neural networks and learning machines"*, 3/E. Pearson Education India, 2011. https://www.pearson.com/en-us/subject-catalog/p/Haykin-Neural-Networks-and-Learning-Machines-3rd-Edition/P200000003278/9780133002553 2. George J. Klir and Bo Yuan, "*Fuzzy Sets and Fuzzy Logic- Theory and Applications*", Prentice Hall of India, 2015. https://www.worldcat.org/title/fuzzy-sets-and-fuzzy-logic-theory-and-applications/oclc/505215200					
	References: 1. Shivanandam, Deepa S, "*Principles of Soft computing*", N Wiley India, 3rd Edition, 2018.https://www.wileyindia.com/principles-of-soft-computing-3ed.html 2. Timothy J. Ross, "*Fuzzy Logic with Engineering Applications*", Third Edition, Wiley, 2011. https://onlinelibrary.wiley.com/doi/book/10.1002/9781119994374 3. Kumar S., "*Neural Networks - A Classroom Approach*", Tata McGraw Hill, 2nd Edition 2017.https://www.worldcat.org/title/neural-networks-a-classroom-approach/oclc/56955342 4. Fakhreddine O. Karray, and Clarence W. De Silva. "*Soft computing and intelligent systems design: theory, tools, and applications*". Pearson Education, 2009. **Weblinks** https://www.pearson.com/en-gb/search.html?q=Karray%20Soft-Computing-and-Intelligent-Systems-Design-Theory-Tools-and-Applications					

	Topics relevant to "Skill Development": Assignment implementations in software, batch wise presentationsare usedforSkill Development through Participative Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE3069	**Course Title: Introduction to Bioinformatics** **Type of Course: General CSE Basket, Theory based**		**L- T- P- C**	3	0	0	3
Version No.	2.0						
Course Pre-requisites	NIL						
Anti-requisites	NIL						
Course Description	This course is designed to provide the knowledge of the concepts related to bioinformatics. The course is aimed at understanding the DNA and Protein sequences and databases. It also deals with Pairwise comparison and calculating the scoring matrix. Further, it focuses on Sequence Alignment techniques, discovering the Motifs in the sequence. Students will also learn the overview of Structural Bioinformatics and Genome sequencing.						
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Introduction to Bioinformatics** and attain **Employability** through **Participative Learning** techniques.						
Course Outcomes	**C.O.1:** Understand the DNA Protein sequence and structures. (Bloom's Level: **Knowledge)** **C.O.2:** Explain the file formats and sequence alignments of DNA sequence. (Bloom's Level: **Comprehension)** **C.O.3:** Apply the techniques of the motifs discovery for the analysis of Protein Sequence. (Bloom's Level: **Application)**						
Course Content:							
Module 1	**Fundamentals of Bioinformatics**	Quiz	Comprehension based Quizzes and assignments;				**9 Classes**
Topics: **Introduction to Bioinformatics**: Introduction to molecular biology, Cell, DNA, RNA, Transcription, Translation, Folding, Gene Structure, Introduction to Bioinformatics, Components and fields of bioinformatics, Omics, basic principles of structural/functional analysis of biological molecules, Biological Data Acquisition, Types of DNA sequences,Genomic DNA,Mitochondrial DNA,DNA Sequencing tools, Protein sequencing and structure determination methods, Finding Reverse complement of a sequence.							
Module 2	**Genome databases and Sequence Similarity**	Quizzes and assignments	Comprehension based Quizzes and assignments				**8 Classes**
Topics: Types and classification of genome databases, DNA sequence retrieval system, various DNA and protein sequence file formats, Common sequence file formats; Files for multiple sequence alignment; Files for structural data, Frequent words and k-mers in Text, String Reconstruction problem, Sequence Similarity searching, Sequence Similarity searching tools, NCBI BLAST, PSI BLAST, Significance of sequence alignments, Alignment scores and gap penalties.							
Module 3	**DNA sequence analysis and applications**	Quizzes and assignments	Comprehension based Quizzes and assignments				**10 Classes**
Sequence similarity searches and alignment tools, Finding alignment using Needleman-Wunsch and Smith-Waterman algorithm, Heuristic Methods of sequence alignment, Pair-wise and multiple sequence alignments, DNA sequence analysis, Motif in protein sequence ,Motif discovery using Gibbs sampling,Motif finding, Gene Prediction models: Hidden Markov model(HMM), Generalized Hidden Markov model(GHMM), Bayesian method.							
Targeted Application & Tools that can be used: BLAST, FastA, , ClustalW, MEGA							
Project work/Assignment:							
Each batch of students (self-selected batch mates – up to 4 in a batch) will be allocated case studies/assignments							

Textbook(s): 1. Bioinformatics: Sequence and Genome Analysis, David W. Mount, Cold Spring Harbor Laboratory Press, 2004. 2. Introduction to Bioinformatics, Arthur Lesk, Fifth Edition, Oxford University Press, 2019
References 1. Bioinformatics Methods and Applications, S. C. Rastogi, N.Mendiratta, P.Rastogi, Fourth Edition, Prentice Hall India. 2.Bioinformatics Algorithms- An Active Learning Approach, Phillip Compeau & Pavel Pevzner, 2nd Edition, Vol. I & II, Active Learning Publishers, 2015 E-References 1. https://puniversity.informaticsglobal.com:2229/login.aspx
Topics related to development of "Employability skills": Batch wise presentations on selected topics 1. String Reconstruction problem 2. Sequence Similarity searching 3. Alignment scores and gap penalties 4. Protein sequencing 5. Gene Prediction models: Hidden Markov model(HMM) 6. Finding similarities by performing pairwise and multiple sequence alignment, 7. Evaluating phylogenetic trees. fordeveloping **Employability Skills** through **Participative Learning techniques**. This is attained through assessment component mentioned in course handout.

urse Code: E3407	**urse Title: Algorithms in Computational Biology** **pe of Course: Program Core** **eory & Lab Integrated**		**T-P-C**	2	0	2	3
rsion No.						
urse Pre-requisites	•	CSE3069 – Introduction to Bioinformatics				
ti-requisites		L				
urse Description		This course introduces core algorithms used in computational biology to solve biological problems efficiently. It covers sequence analysis, dynamic programming, genome assembly, and string matching techniques. Students will learn how to apply algorithmic thinking to biological datasets and evaluate the performance of solutions.				

urse Object		The objective of the course is to familiarize the learners with the concepts of Algorithams in Computational Biology and attain **Skill Development** through **Experiential Learning** techniques.
urse Out Comes		On successful completion of the course the students shall be able to: **CO1:** Define key concepts in computational biology and bioinformatics. *(Remember)* **CO2:** Explain the significance of algorithms in analyzing biological data. *(Understand)* **CO3**: Apply basic algorithmic strategies to solve simple biological problems. *(Apply)* **CO4:** Explore different types of biological data and their computational needs. *(Apply)*
urse Content:		

odule 1	**Introduction**	signment			**5L+6P Sessions**
	Topics: History and principles of algorithms, Types of algorithms, Development and computational complexity .				
odule 2	gorithms Issues and Problems	signment			**+8P Sessions**
	Topics: Asymptotic analysis of algorithms, NP-complete problems, Polynomial reducibility, Traveling Salesman Problem (TSP), Consecutive Integer Problem (CIP), Sorting problems and Fibonacci problem				
odule 3	gorithmic Approaches	signment			**L+8PSessions**
	Topics: Linear, exhaustive search, branch and bound, divide and conquer, Expectation and Maximization (EM), Forward and backward algorithms, Discriminative learning, Knuth-Morris-Pratt and Boyer-Moore (string matching), Graph algorithms and maximum likelihood algorithms				
odule 4	namic Programming & Methods	signment			**+8P Sessions**
	pics: Principles and applications, Heuristics tools (BLAST, FASTA, ClustalW),				

	Probabilistic/statistical methods, Models of evolution and relevant algorithms, Partial and double digest problems, Graph algorithms for DNA sequence assembly (CASP3, Phrap, Phred), Protein structure prediction – Chou-Fasman algorithm.
	ject work/Assignment:
3.	**4. Assignment 1 on (Module 1 and Module 2)** 5. **Assignment 2 on (Module 3 and Module 4)**
	Text Book 1) **Phillip Compeau & Pavel Pevzner, *Bioinformatics Algorithms: An Active Learning Approach*, Vol. 1, 2nd Ed., 2019**. 2) **Michael T. Goodrich & Roberto Tamassia, *Algorithm Design and Applications*, Wiley, 2015**. 3) **Jason Kinser,** *Computational Biology: A Hypertextbook*, 2nd Ed., Jones & Bartlett Learning, 2021 4) **Gautam B. Singh**, *Fundamentals of Bioinformatics and Computational Biology*, Springer, 2015.
	References 1. Zhumur Ghosh & Bibekanand Mallick, *Bioinformatics: Principles and Applications*, Oxford University Press, 2014. 2. Steven Skiena, *The Algorithm Design Manual*, Springer, 2nd Ed., 2008 3. Pavel Pevzner, Computational Molecular Biology: An Algorithmic Approach, MIT Press, 2000. 4. T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, *Introduction to Algorithms*, 3rd Ed., MIT Press, 2009. 5. Arthur Lesk, *Introduction to Bioinformatics*, 5th Ed., Oxford University Press, 2019 **Web Resources** W1. https://onlinecourses.nptel.ac.in/noc25_cs06/preview W2. https://ocw.mit.edu/courses/6-096-algorithms-for-computational-biology-spring-2005/pages/lecture-notes/
	Module I: Introduction to Algorithms **Experiment 1:** ***Implement Sorting and Searching Algorithms*** • **Implement Bubble, Merge Sort, Linear and Binary Search** • **Compare time complexity using real biological data (e.g., gene lengths)** **Experiment 2:** ***Time Complexity and Recursion using Fibonacci Series*** • **Compare recursive and dynamic programming approaches**

	• **Visualize time/memory usage with time and memory_profiler modules** **Module II: Algorithmic Problem Solving** **Experiment 3:** ***Solve the Travelling Salesman Problem (TSP)*** o **Use brute force or greedy algorithms** o **Simulate sequencing fragment reassembly as a path problem** **Experiment 4:** ***Knapsack Problem in Bioinformatics*** o **Apply knapsack logic to protein interaction weighting or resource allocation** **Module III: String Matching and Sequence Analysis** **Experiment 5:** ***Naive Pattern Matching Algorithm on DNA Sequence*** • **Identify motifs like start/stop codons** **Experiment 6:** ***KMP and Boyer-Moore Algorithms*** • **Compare performance on large FASTA datasets** **Experiment 7:** ***Regular Expression Matching in Genomic Data*** • **Search for specific motifs** **Module IV: Dynamic Programming** **Experiment 8:** ***BLAST Query using Biopython (Online)*** **Submit a BLAST query and parse top hits** **Experiment 9:** ***DNA Read Assembly Simulation***
	Topics relevant to development of "Employability": Proficiency in bioinformatics algorithms, data analysis **Topics relevant to "PROFESSIONAL ETHICS":** Maintaining professional integrity in computational biology.

urse Code: E3408	**urse Title: Statistical Methods for BioInformatics** **pe of Course: Program Core** **eory & Lab Integrated**	**Γ-P-C**	2	0	2	3

rsion No.		
urse Pre-requisites	•	CSE3069– Introduction to Bioinformatics
ti-requisites		L
urse Description		This course provides an introduction to the statistical methods commonly used in bioinformatics and biological research. The course briefly reviews basic probability and statistics including events, conditional probabilities, Bayes theorem, random variables, probability distributions, and hypothesis testing and then proceeds to topics more specific to bioinformatics research, including Markov chains, hidden Markov models, Bayesian statistics, and Bayesian networks. Students will learn the principles behind these statistical methods and how they can be applied to analyze biological sequences and data..
urse Object		The objective of the course is to familiarize the learners with the concepts of Statistical Methods for BioInformatics and attain **Skill Development** through **Experiential Learning** techniques.
urse Out Comes		On successful completion of the course the students shall be able to: **CO1:** Understand the basic concepts of bioinformatics including databases. Understand) **CO2:** Evaluate methods to characterize and manage the different types of biological sequence data. (Apply) **CO3**: Apply basic concepts in biostatistics exemplifying sampling methods, (Apply) **CO4:** Evaluate hypothesis testing using statistical methods(Apply)
urse Content:		

ɔdule 1	Bioinformatics Fundamentals and Database Systems	signment			**6L+6P Sessions**
	Topics:				

	Bioinformatics definition, history, scope and applications, Bioinformatics web portals: NCBI, EBI, ExPASy, Biological databases: Classification of databases - primary (Genbank), secondary (PIR) and tertiary or composite (KEGG) databases, Sequence databases - DNA sequence databases (ENA,DDBJ), Protein sequence databases (Swissprot, PROSITE) .				
odule 2	quence Alignment	signment			+8P Sessions
	Topics: Basics of sequence alignment - match, mismatch, gaps, gap penalties, scoring alignment, Types of sequence alignment - pairwise and multiple alignment, local and global alignment, Dot matrix comparison of sequences, Scoring matrices - PAM and BLOSUM, Pairwise sequence similarity search by BLAST and FASTA				
odule 3	sic Concepts In Biostatistics	signment			L+8PSessions
	Topics: Introduction to Biostatistics, kinds of data and variables - based on nature (numericaldiscrete and continuous, categorical-ordinal and nominal) - based on source (primary and secondary data), sample size, sampling methods and sampling errors, Data tabulation and representation methods: graphical methods– stem and leaf plot, line diagram, bar graphs, histogram, frequency polygon, frequency curves; diagrammatic method- pie diagram, Measures of central tendency- mean, median, mode; merits and demerits, Measures of dispersion- range, variance, standard deviation, standard error and coefficient of variation; merits and demerits, Correlation and regression analysis and their applications to biolog.				
odule 4	ostatistics-Applications	signment			+8P Sessions
	pics: roduction to Probability- definition; Normal distribution: definition and properties, Hypothesis testing- steps in testing for statistical hypothesis, null and alternative hypothesis, level of significance- type-1 and type-2 errors, Test of significance for large samples- Z-test for means and proportions, Test of significance for small samples- student's t-test(one sample and two samples), Chi-square test and its applications- goodness of fit (not based on distribution), test of independence , Analysis of variance (One-way ANOVA) and their applications to biology .				
	oject work/Assignment:				
6.	**7. Assignment 1 on (Module 1 and Module 2)** 8. **Assignment 2 on (Module 3 and Module 4)**				
	Text Book 5) **Arthur Lesk**, *Introduction to Bioinformatics*, Oxford University Press, 5th Edition, 2019. 6) **Wayne W. Daniel & Chad L. Cross**, *Biostatistics: A Foundation for Analysis in the Health Sciences*, Wiley, 11th Edition, 2019. 7) **Gautam B. Singh**, *Fundamentals of Bioinformatics and Computational Biology*, Springer, 2015.				

	References **1.** Zhumur Ghosh & Bibekanand Mallick, *Bioinformatics: Principles and Applications*, Oxford University Press, 2014. **2.** David W. Mount, *Bioinformatics: Sequence and Genome Analysis*, Cold Spring Harbor, 2nd Edition, 2004**.** **3.** Marcello Pagano & Kimberlee Gauvreau, *Principles of Biostatistics*, CRC Press, 2nd Edition, 2018. 4. **S. C. Gupta & V. K. Kapoor**, *Fundamentals of Mathematical Statistics*, Sultan Chand **Web Resources** W3. https://www.ncbi.nlm.nih.gov W4. https://archive.nptel.ac.in/courses/102/101/102101056/ W5. https://onlinecourses.nptel.ac.in/noc25_bt06/preview W6. https://www.ebi.ac.uk W7. https://www.expasy.org
	Module I: Bioinformatics & Biological Databases 1. **Experiment 1: Introduction to NCBI – Searching for DNA & protein sequences** 2. **Experiment 2: Exploring the EBI and ExPASy portals** 3. **Experiment 3: Retrieving gene information from GenBank and ENA** 4. **Experiment 4: Identifying protein domains using PROSITE and SwissProt** **Module II: Sequence Alignment** 5. **Experiment 5: Pairwise sequence alignment using EMBOSS Needle** 6. **Experiment 6: Local sequence alignment using BLAST** 7. **Experiment 7: Global sequence alignment using Clustal Omega** 8. **Experiment 8: Constructing phylogenetic trees using MEGA or Phylogeny.fr** **Module III: Basic Biostatistics** 9. **Experiment 9: Data collection and classification of variables** 10. **Experiment 10: Creating bar charts, pie charts, histograms using MS Excel or Python (matplotlib/seaborn)**

	11. **Experiment 11: Calculating mean, median, mode, standard deviation using statistical software (R/SPSS/Excel)** **Module IV: Statistical Analysis** 12. **Experiment 12: Performing correlation and regression analysis in R** 13. **Experiment 13: Hypothesis testing using t-test and Z-test in SPSS/R** 14. **Experiment 14: Conducting chi-square test for independence** 15. **Experiment 15: One-way ANOVA application on biological data s**
	Topics relevant to development of "Employability": Training in bioinformatics tools, biological databases, sequence analysis **Topics relevant to "PROFESSIONAL ETHICS":** Data privacy in genomic research, ethical use of biological databases

Course Code: CSE3409	Course Title: Emerging Technologies in Big Data	L-T-P- C	2 -0	2	3
Version No.	1.0				
Course Pre-requisites	CSE3156-Database Management System,				
Anti-requisites	NIL				
Course Description	The purpose of the course is to provide the fundamentals of Big data technology, to emphasize the importance of choosing suitable tools for processing and analyzing big data to gain insights. The student should have knowledge and skill to select and use most appropriate big data tools to solve business problems. The associated laboratory provides an opportunity to implement the concepts and enhance critical thinking and analytical skills. With a good knowledge in the fundamentals of Big data technology the student can gain practical experience in implementing them, enabling the				

	student to be an effective solution provider for applications that involve huge volume of data.			
Course Objectives	The objective of the course is to familiarize the learners with the concepts of Big Data Technologies and attain SKILL DEVELOPMENT through EXPERIENTIAL LEARNING techniques.			
Course Outcomes	On successful completion of the course the students shall be able to: Apply Map-Reduce programming on the given datasets to extract required insights. (Application). Employ appropriate Hadoop Ecosystem tools such as scoop, Hbase, Hive, to perform data analytics for a given problem. (Application). Use Spark tool to analyze the given dataset for a given problem. (Application).			
Course Content:				
Module 1	Introduction to Hadoop	Programming Assignment	Data Collection and Analysis	10 Classes
Introduction to Big Data and its importance: Basics of Distributed File System, Four Vs, Drivers for Big data, Big data applications, Structured, unstructured, semi-structured and quasi structured data. Big data Challenges-Traditional versus big data approach, The Big Data Technology Landscape: No-SQL. The Hadoop: History of Hadoop-Hadoop use cases, The Design of HDFS, Blocks and replication management, Rack awareness, HDFS architecture, HDFS Federation, Name node and data node, Anatomy of File write. Anatomy of File read, Hadoop Map Reduce paradigm, Map and reduce tasks, Job Tracker and task tracker, Map reduce execution pipeline, Key value pair, Shuffle and sort, Combiner and Partitioner, APIs used to Write/Read files into/from Hadoop, Need for Flume and Sqoop. Anatomy of a YARN: Hadoop 2.0 Features, Name Node High Availability, YARN Architecture, Introduction to Schedulers, YARN scheduler policies, FIFO, Fair And Capacity scheduler.				
Module 2	Hadoop Ecosystem Tools	Programming Assignment	Data Collection and Analysis	8 Classes
Introduction to SQOOP: SQOOP features, Sqoop Architecture, Sqoop Import All Tables, Sqoop Export All Tables, Sqoop Connectors, Sqoop Import from MySQL to HDFS, Sqoop vs flume. Hive: Apache Hive with Hive Installation, Hive Data Types, Hive Table partitioning, Hive DDL commands, Hive DML commands, and Hive sort by vs. order by, Hive Joining tables, Hive				

bucketing.

Hbase: Introduction to HBase and its working architecture- Commands for creation and listing of tables- disabled and is disabled of table - enable and is enabled of table- describing and dropping of table-Put and Get command - delete and delete all command-commands for scan, count, truncate of tables.

Module 3	Spark	Programming Assignment	Data analysis	8 Classes

Introduction to Apache Spark A unified Spark, Who uses Spark and for what?, A Brief History of Spark, Spark version and releases, Storage layers for Spark. Programming with RDDs: RDD Basics, Creating RDDs, RDD Operations, Passing functions to Spark, Common Transformations and Actions, Persistence. Spark SQL: Linking with Spark SQL, Using Spark SQL in Applications, Loading and Saving Data, JDBC/ODBC Server, User-defined functions, Spark SQL Performance.

Scala: The Basics, Control Structures and functions, Working with arrays, Maps and Tuples.

List of Laboratory Tasks:

1. Level 1: To install the Hadoop in pseudo cluster mode.

   Level 1: HDFS Shell Commands – Files and Folders.

   Level 2: HDFS Shell Commands – Management.

2. Run a basic Word Count Map Reduce program to understand Map Reduce Paradigm.

   Level 1: Find the number of occurrence of each word appearing in the input file(s)

   Level 2: Performing a Map Reduce Job for word search count (look for specific keywords in a file).

3. Write a Map Reduce program that mines weather data. Weather sensors collecting data every hour at many locations across the globe gather large volume of log data, which is a good candidate for analysis with Map Reduce, since it is record-oriented. Data available at: https://github.com/tomwhite/hadoopbook/tree/master/input/ncdc/all.

   Level 1: Find average, max and min temperature for each year in NCDC data set?

   Level 2: Programming assignment to analyze the social media data for business analytics.

4. Level 1: Finding out Number of Products Sold in Each Country using map reduce with sample dataset

   Level 2: Find matrix multiplication using map reduce

5. Level 1: Installation of Hive, working on basic hive commands. (Create, Alter and Drop tables)
   Level 2: Apply Hive commands to student database/employee database.

6. Level 1: Working on advance hive commands. (Static Partitioning & Dynamic partitioning)
   Level 2: Continue the previous experiment, select and apply suitable partitioning technique.

7. Level 1: Working on advance hive commands-2. (Bucketing)
   Level 2: Continue the previous experiment, apply bucketing technique to bring out the difference between partitioning and bucketing.

8. Level 1: Installing Ecosystem tools such as Scoop, Hbase.
   Level 2: Scoop – Move Data into Hadoop.

9. Level 1: Working on basic Hbase commands (General commands, DDL Commands)
   Level 2: Apply Hbase commands on Insurance database/employee dataset.

10. Level 1: Working on advanced Hbase commands. (DML).
    Level 2: Continue the previous experiment to demonstrate CRUD operations.

11. Level 1: Install, Deploy & configure Apache Spark.
    Level 2: Using RDD and FlatMap count how many times each word appears in a file and write out a list of words whose count is strictly greater than 4 using Spark

12. Level 1: Write a program in Apache spark to count the occurrences words in a given text file and display only those words starting with 'a' in ascending order of count.
    Level 2: Apache access logs are responsible for recording data for all web page requests processed by the Apache server. An access log record written in the Common Log

Format will look something like this: 127.0.0.1 - Scott [10/Dec/2019:13:55:36 – 0700] "GET /server-status HTTP/1.1" 200 2326 Where, HTTP 200 status response code indicates that the request has succeeded. Write a program to read the records of access log file log.txt and display the number of successful requests using Spark.

13. Level 1: Chess king moves horizontally, vertically or diagonally to any adjacent cell. Given two different cells of the chessboard, determine whether a king can go from the first cell to the second in one move.

Write a scala program that receives input of four numbers from 1 to 8, each specifying the column and row number, first two - for the first cell, and then the last two - for the second cell. The program should output YES if a king can go from the first cell to the second in one move, or NO otherwise.

Level 2: Data analytics using Apache Spark on Amazon food dataset, find all the pairs of items frequently reviewed together.

Write a single Spark application that:

Transposes the original Amazon food dataset, obtaining a Pair RDD of the type:

Counts the frequencies of all the pairs of products reviewed together;

Writes on the output folder all the pairs of products that appear more than once and their frequencies. The pairs of products must be sorted by frequency.

Targeted Application & Tools that can be used:

Business Analytical Applications

Social media Data Analysis

Predictive Analytics

Tools: Hadoop Framework tools like map reduce, Hive, Hbase, Scoop, Spark.

Text Book

Seema Acharya, Subhashini Chellappan. 2015. Big Data and Analytics. Wiley Publication.

Matei Zaharia, Bill Chambers. 2018. SPARK: The Definitive Guide. Oreilly.

References
Tom White. 2016. Hadoop: The Definitive Guide. O'Reilley. Cay S. Horstmann. 2017. Scala for the Impatient. Wesley.
Topics relevant to development of "Skill Development": Real time application development using Hadoop Ecosystem tools through Experiential Learning as mentioned in the course handout.

Course Code: CSE3410	**Course Title: Statistical Techniques for Data Science** **Type of Course: Theory**	**L-T-P-C**	2	0 2	3
Version No.	1.0				
Course Pre-requisites	MAT1003				
Anti-requisites	NIL				
Course Description	This course provides an in-depth introduction to statistics and machine learning theory, methods, and algorithms for data science. Topics include multiple regression, kernel learning, sparse regression, generalized linear models, supervised and unsupervised learning, deep learning, covariance learning, factor models, principal component analysis, and more. The course emphasizes the applicability and limitations of these methods using mathematical statistics and real-world data sets.				
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Statistical Foundations for Data Science** and attain **Employability** through **Participative Learning** techniques.				
Course Out Comes	On successful completion of the course the students shall be able to: 4) Understand the rise and significance of Big Data in various fields such as Biological Sciences, Health Sciences, Computer and Information Sciences, Economics and Finance, Business and Program Evaluation, Earth Sciences, and Astronomy. 4) Develop a strong foundation in multiple linear regression and the Gauss-Markov theorem.				

	4) Apply linear regression with random design and partial linear regression. 4) Apply the power method and learn about factor models and structured covariance learning.			
Course Content:				
Module 1	Introduction	Assignment	Programming	**No. of Classes:10**
Topics: Introduction to bigdata, Rise of Big Data and Dimensionality in -Biological Sciences ,Health Sciences , Computer and Information Sciences , Economics and Finance, Business and Program Evaluation, Earth Sciences and Astronomy - Impact of Big Data - Impact of Dimensionality , Computation of Noise Accumulation , Spurious Correlation , Statistical theory - Aim of High-dimensional Statistical Learning.				
Module 2	Multiple Linear Regression	Assignment	Programming	**No. of Classes:12**
Topics: Multiple Linear Regression, The Gauss-Markov Theorem , Statistical Tests - Weighted Least-Squares , Box-Cox Transformation , Model Building and Basis Expansions, Polynomial Regression - Spline Regression , Multiple Covariates , Ridge Regression - Bias-Variance Tradeoff - Penalized Least Squares - Bayesian Interpretation - Ridge Regression Solution Path - Kernel Ridge Regression , Exponential family 231 5.1.2 Elements of generalized linear models , Maximum likelihood , Computing MLE: Iteratively reweighed least squares , Deviance and Analysis of Deviance, Regularization parameters, Refitted Cross-validation, Extensions to Nonparametric Modeling.				
Module 3	Inference in linear regression	Assignment	Programming	**No. of Classes:14**

Topics:

Inference in linear regression - Debias of regularized regression estimators , Choices of weights , Inference for the noise level , Inference in generalized linear models , Desparsified Lasso , Decorrelated score estimator - Test of linear hypotheses , Numerical comparison - Asymptotic efficiency 345 7.3.1 Statistical efficiency and Fisher information, Linear regression with random design , Partial linear regression , Gaussian graphical models - Inference via penalized least squares , Sample size in regression and graphical models , General solutions , Local semi-LD decomposition , Data swap , Gradient approximation

Module 4	Principal Component Analysis	Assignment	Programming	**No. of Classes:9**

Topics:

Principal Component Analysis -Introduction to PCA , Power Method , Factor Models and Structured Covariance Learning , Factor model and high-dimensional PCA-Cluster Analysis - K-means clustering , Hierarchical clustering , Model-based clustering , Spectral clustering , Data-driven choices of the number of clusters , Variable Selection in Clustering , Sparse K-means clustering , Sparse model-based clustering , Sparse Mixture of Experts Model, Correlation Screening, Generalized and Rank Correlation Screening, Nonparametric Screening, Sure Screening and False Selection.

Targeted Application & Tools that can be used:

Tools: Torch, Google Colaboratory, Spider, Jupiter Notebook

Project work/Assignment:

Text Book

TextBook(s):

T1 Fan, J., Li, R., Zhang, C.-H., and Zou, H. (2020). Statistical Foundations of Data Science. CRC Press.

T2 Wainwright, M. J. (2019). *High-dimensional statistics: A non-asymptotic viewpoint.* Cambridge University Press.

References

R1. James, G., Witten, D., Hastie, T.J., Tibshirani, R. and Friedman, J. (2013). *An Introduction to Statistical Learning with Applications in R* . Springer, New York.

R2. Hastie, T.J., Tibshirani, R. and Friedman, J. (2009). *The elements of Statistical*

Learning: Data Mining, Inference, and Prediction (2nd ed). Springer, New York.

R3. Buehlmann, P. and van de Geer, S. (2011). *Statistics for High-Dimensional Data: Methods, Theory and Applications*. Springer, New York.

Book link

R1: Fan, J., Li, R., Zhang, C.-H., and Zou , Statistical Foundations of Data Science. CRC Press.

E book link

R2: W. N. Venables, D. M. Smith and the R Core Team, https://cran.r-roject.org/doc/manuals /R-intro.pdf, October,2022

Web resources:

W1. https://www.youtube.com/playlist?list=PLOU2XLYxmsIK9qQfztXeybpHvru-TrqAP

https://presiuniv.knimbus.com/user#/

Topics relevant to "EMPLOYABILITY SKILLS": - Asymptotic efficiency 345 7.3.1 Statistical efficiency and Fisher information, Linear regression with random design , Partial linear regression **for developing Employability Skills through Participative Learning techniques. This is attained through assessment component mentioned in course handout..**

Catalogue prepared by	
Recommended by the Board of Studies on	
Date of Approval by the Academic Council	

Course Code: CSE3411	**Course Title:** Predictive Analytics and Applications **Type of Course: Program Core**		**L-P-C**	2	0 2	3
Version No.	1					
Course Pre-requisites	MAT1003					
Anti-requisites	NIL					
Course Description	Predictive Analytics subject is conceptual in nature. The students will be benefited in this course to know about modern data analytic concepts and develop the skills for analyzing and synthesizing data sets for decision making in the firms.					
Course Objective	The objective of the course is skill development of student by using Learning techniques					
Course Out Comes	**On successful completion of the course the students shall be able to:** • CO 1: **Define** the nature of analytics and its applications. (Remember) • CO 2: **Summarize** the concepts of predictive analytics and data mining.(Understand) • CO 3: **Construct** the analytical tools in business scenarios to achieve competitive advantage.(Apply) • CO 4: **Build** the real-world insights in decision trees and time series analysis methods in dynamic business environment.(Apply)					
Course Content:						
Module 1	Introduction to Predictive Analytics	Self-Learning	Applications of analytics			**7 Sessions**
Topics: Analytics- Definition, importance, Analytics in decision making, Applications, Challenges, Experts perception on analytics; Popularity in Analytics; Predictive analytics in business Scenarios- case studies						
Module	**Principles and**	Case				**8 Sessions**

2	**Techniques**	analysis		
Topics: Predictive modeling: Propensity models, cluster models, collaborative filtering, applications and limitations - Statistical analysis: Univariate Statistical analysis, Multivariate Statistical analysis				
Module 3	**Model Selection**	Participative Learning & Case Analysis		**7 Sessions**
Preparing to model the data: supervised versus unsupervised methods, statistical and data mining methodology, cross-validation, overfitting, bias-variance trade-off, balancing the training dataset, establishing baseline performance. Measuring Performance in Regression Models - Linear Regression and Its Cousins - Non-Linear Regression Models - Regression Trees and Rule-Based Models Measuring Performance in Classification Models - Discriminant Analysis and Other Linear Classification Models - Non-Linear Classification Models				
Module 4	**Time Series Analysis**	Discussion & Presentation		8 Sessions
Time series Model: ARMA, ARIMA, ARFIMA - Temporal mining - Box Jenkinson method, temporal reasoning, temporal constraint networks				

Text Book

1.Jeffrey Strickland, Predictive analytics using R, Simulation educators, Colorado Springs, 2015

2.Max Kuhn and Kjell Johnson, Applied Predictive Modeling, 1st edition Springer, 2013.

References

R1 Dinesh Kumar, U. (2021). Business Analytics: The Science of data-Driven Decision Making.

R2 Business Analytics - Data Analysis & Decision Making", S. Christian Albright and Wayne L. Winston, Cengage Publication, 5th Edition, 2012

E book link R1: Raman, R., Bhattacharya, S., & Pramod, D. (2018). Predict employee attrition by using predictive analytics. Benchmarking: An International Journal. https://www-emerald-com-presiuniv.knimbus.com/insight/content/doi/10.1108/BIJ-03-2018- 0083/full/html

2. **E book link R2:** Jing, Z., Luo, Y., Li, X., & Xu, X. (2022). A multi-dimensional city data embedding model for improving predictive analytics and urban operations. Industrial Management & Data Systems, (ahead-of-print). https://www-emerald-com-presiuniv.knimbus.com/insight/content/doi/10.1108/IMDS-01-2022- 0020/full/html
3. **E book link R3:** Singh, R., Sharma, P., Foropon, C., & Belal, H. M. (2022). The role of big data and predictive analytics in the employee retention: a resource-based view. International Journal of Manpower. https://www-emerald-com-presiuniv.knimbus.com/insight/content/doi/10.1108/IJM-03-2021- 0197/full/html
4. **E book link R4:** Mishra, D., Luo, Z., Hazen, B., Hassini, E., & Foropon, C. (2018). Organizational capabilities that enable big data and predictive analytics diffusion and organizational performance: A resource-based perspective. Management Decision. https://www-emerald-com-presiuniv.knimbus.com/insight/content/doi/10.1108/MD-03-2018- 0324/full/html

Web resources:

W1.https://www.sas.com/en_in/insights/analytics/predictive-analytics.html

W2. https://www.techtarget.com/searchbusinessanalytics/definition/predictive-analytics

W3. https://www.cio.com/article/228901/what-is-predictive-analytics-transforming-data- intofuture-insights.html

W4. https://www.simplilearn.com/what-is-predictive-analytics-article

W5. https://www.northeastern.edu/graduate/blog/predictive-analytics/

W6.https://www.marketingevolution.com/knowledge-center/the-role-of-predictive-analyticsin-data-driven-marketing

Swayam & NPTEL Video Lecture Sessions on Predictive Analytics

1. https://onlinecourses.swayam2.ac.in/imb20_mg19/preview
2. https://onlinecourses.nptel.ac.in/noc19_mg42/preview

Case References

1. Predictive Analytics Industry Use cases.
2. https://www.rapidinsight.com/blog/11-examples-ofpredictive-analytics/
3. Srinivasan Maheswaran (2017). Predictive Analytics – Employee Attrition Case center.

Topics relevant to development of "Skill Development": ": Application of Business Analytics to enhances customer satisfaction and firms' success

Topics relevant to development of "Environment and sustainability: Focus on Predictive analytics to minimize the errors in decision making

Course Code: CSE2021	**Course Title: Data Mining** **Type of Course: Discipline Elective/ Theory Only Course**	**L- T-P-C**	3	0	0	3

Version No.		2.0
Course Pre-requisites		MAT1003 – Applied Statistics
Anti-requisites		**NIL**
Course Description		Introduction, Applications, issues in data mining, data pre-processing techniques, data mining tasks, association rules, advanced association rules, classification, different approaches for classification, clustering, outlier detection. Recent trends in data mining.
Course Objective		The objective of the course is to familiarize the learners with the concepts of **Data Mining** and attain **Employability** through **Problem Solving** Methodologies
Course Out Comes		**On successful completion of the course the students shall be able to:** • Apply the various pre-processing techniques needed for a data mining task. • Understand the functionality of the various data mining algorithms. • Appreciate the strengths and limitations of various data mining models. • Understand the advances in data mining for real life applications.
Course Content:		

Module 1	**Introduction to Data Mining**	Assignment		Data Collection	**5 Sessions**
	Topics: Introduction to Data mining – Data Mining Goals– Stages of the Data Mining Process–Data Mining Techniques– Merits and Demerits.				
Module 2	**Data preprocessing**	Quiz		Problem Solving	**9 Sessions**
	Topics: Types of data – Pre Processing steps – Data Preprocessing Techniques – Similarity and Dissimilarity measures.				
Module 3	**Data Mining – Frequent Patterns**	Assignment		Problem Solving	**7 Sessions**
	Topics: Market Basket Analysis, item sets – Generating frequent item sets and rules efficiently – Apriori Algorithm– FPGrowth.				
Module 4	**Classification and clustering**	Assignment		Problem Solving	**11 Sessions**
	Classification and Clustering Decision tree Induction – Bayesian classification –Classification by Back Propagation - Lazy learners – Modern evaluation and selection techniques to improve classification accuracy. Clustering Analysis – portioning method – Hierarchical methods – Density based method				
Module 5	**Outlier detection & Data**	Assignment		Problem Solving	**5 Sessions**

	mining trends					

Anomaly detection preliminaries - Different Outlier detection techniques-Web mining- Text mining- Demonstration of Weka tool.

Project work/Assignment:

Assignments

1. From the dataset given, find the Entropy, Gain value of the attributes and also draw the decision tree using entropy for the given dataset.

2. Transactional Data Base, D given below which contains set of items find the frequent item set using the Apriori Algorithm and generate the Association Rules. Minimum Support count is 2%. Minimum confidence is 60%.

T_{id}	Items
10	1, 3, 4
20	2, 3, 5
30	1, 2, 3, 5
40	2, 5

Text Book

T1 T1. Tan P. N., Steinbach M & Kumar V. "Introduction to Data Mining" ,Pearson Education, 2016.

References

R1 *Han J & Kamber M, "Data Mining: Concepts and Techniques", Elsevier, Second Edition, 2006*

R2 *G K Gupta, "Introduction to Data Mining with Case Studies", PHI, Third Edition, 2014.*

R3 *Alex Berson and Stephen J. Smith, "Data Warehousing, Data Mining and OLAP", Tata McGraw – Hill*

Additional web-based resources

W1. https://onlinecourses.swayam2.ac.in/cec20_cs12/previewText book of Data Mining: Concepts and Techniques, Jiawei Han, Micheline Kamber and Jian Pei, Morgan Kaufmann Publishers, 2012.
W2.https://puniversity.informaticsglobal.com:2284/ehost/detail/detail?vid=7&sid=e2d7362a-fd3049a98f0393e963521dbd%40redis&bdata=JnNpdGU9ZWhvc3QtbGl2ZQ%3d%3d#AN=377411&db=nlebk

3. https://nptel.ac.in/courses/105105157

Topics relevant to "EMPLOYABILITY SKILLS": Data Mining Techniques, FP Growth for developing **Employability Skills** through **Participative Learning** techniques. This is attained through the assessment component mentioned in the course handout.

Course Code: CSE3413	**Course Title: No SQL Data Management** **Type of Course: Program Core**	**L-T-P-C**	2	0	2	3
Version No.	1.0					
Course Pre-requisites	CSE3156 – Database Management System					
Anti-requisites	NIL					

Course Description		The **NoSQL Data Management** course provides a comprehensive understanding of non-relational database systems, emphasizing their emergence as scalable and flexible alternatives to traditional relational databases. It covers various NoSQL data models, including key-value, document, column-family, and graph databases, exploring their structure, use cases, and design principles. Students will learn about the challenges of data persistence, concurrency, and integration, and how aggregate-oriented models address these issues. The course delves into distribution models such as sharding and replication, the implications of the CAP theorem, and consistency trade-offs. Additionally, it introduces Map-Reduce for large-scale data processing and guides students in modeling data for access efficiency. Practical applications and limitations of each NoSQL type are discussed, preparing students to design robust, scalable data solutions for real-world applications.
Course Objectives		The objective of the course is to introduce students to the principles, models, and practical applications of NoSQL databases. It aims to equip students with the knowledge and skills necessary to understand the advantages of NoSQL over traditional relational databases, comprehend various data models (key-value, document, column-family, and graph), explore distribution models for scalability and availability, and apply Map-Reduce and other techniques for data processing.
Course Out Comes		1. Understand the limitations of relational databases and explain the motivation behind the emergence of NoSQL databases. 2. Differentiate between various NoSQL data models such as key-value, document, column-family, and graph databases based on structure, use cases, and performance. 3. Analyze and apply appropriate distribution strategies including sharding, replication, and consistency models in distributed NoSQL systems. 4. Design and implement data access patterns using aggregate-oriented modeling and schema-less approaches for scalable NoSQL applications. 5. Utilize Map-Reduce and other data processing techniques to handle large-scale data operations efficiently in NoSQL environments.
Course Content:		

Module 1	**Introduction to NoSQL and Aggregate-Oriented Data Models**	**Quiz**		**Knowledge based quiz**	**No. of sessions:8**
	Why NoSQL? The Value of Relational Databases, Getting at Persistent Data, Concurrency, Integration, A (Mostly) Standard Model, Impedance Mismatch, Application and Integration Databases, Attack of the Clusters, The Emergence of NoSQL, Aggregate Data Models; Aggregates, Example of Relations and Aggregates, Consequences of Aggregate Orientation, Key-Value and Document Data Models, Column-Family Stores, Summarizing AggregateOriented Databases. More Details on Data Models; Relationships, Graph Databases, Schema less Databases, Materialized Views, Modelling for Data Access.				
Module 2	**Distributed Data Systems and Consistency Models**	**Assignment**		**Data Visualization**	**No. of sessions:10**
	Distribution Models; Single Server, Sharding, Master-Slave Replication, Peer-to-Peer Replication, Combining Sharding and Replication Consistency, Update Consistency, Read Consistency, Relaxing Consistency, The CAP Theorem, Relaxing Durability, Quorums. Version Stamps, Business and System Transactions, Version Stamps on Multiple Nodes				
Module 3	**Key-Value Stores and Map-Reduce Framework**	**Design an algorithm using Example**		**Random Forest**	**No. of sessions:10**

	Map-Reduce, Basic Map-Reduce, Partitioning and Combining, Composing Map-Reduce Calculations, A Two Stage Map-Reduce Example, Incremental Map-Reduce Key-Value Databases, What Is a Key-Value Store, Key-Value Store Features, Consistency, Transactions, Query Features, Structure of Data, Scaling, Suitable Use Cases, Storing Session Information, User Profiles, Preference, Shopping Cart Data, When Not to Use, Relationships among Data, Multi operation Transactions, Query by Data, Operations by Sets.					
Module 4	**Document-Oriented Databases and Use Cases**	**Case Study**		**Conduct a case study on how data sets can be gathered and implemented in real time application.**	**No. of sessions:10**	
	Document Databases, What Is a Document Database?, Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms, Web Analytics or Real-Time Analytics, E- Commerce Applications, When Not to Use, Complex Transactions Spanning Different Operations, Queries against Varying Aggregate Structure					
Module 5	**Graph Databases and Connected Data Solutions**	**Case Study**			**No. of sessions:10**	
	Graph Databases, What Is a Graph Database?, Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Connected Data, Routing, Dispatch, and Location-Based Services, Recommendation Engines, When Not to Use.					
	TEXTBOOKS: 1. Sadalage, P. & Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Pearson Addision Wesley, 2012 REFERENCE BOOKS: 1. Dan Sullivan, "NoSQL For Mere Mortals", 1st Edition, Pearson Education India, 2015. (ISBN- 13: 978-9332557338) 2. Dan McCreary and Ann Kelly, "Making Sense of NoSQL: A guide for Managers and the Rest of us", 1st Edition, Manning Publication/Dreamtech Press, 2013. (ISBN-13: 978-9351192022) 3. Kristina Chodorow, "Mongodb: The Definitive Guide- Powerful and Scalable Data Storage", 2nd Edition, O'Reilly Publications, 2013. (ISBN-13: 978-9351102694) VIDEO LINKS: 1. https://www.geeksforgeeks.org/introduction-to-nosql/ (and related links in the page) 2. https://www.youtube.com/watch?v=0buKQHokLK8 (How do NoSQL databases work? Simply explained)					

Course Code: CSE3414	**Course Title: Applied Data Intelligence** **Type of Course: Program Core**	**L-T-P-C**	2	0	2	3
Version No.	1.0					
Course Pre-requisites	Nil					
Anti-requisites	NIL					
Course Description	The aim of the course is to give complete overview of Python's data analytics tools and techniques. Learning python is a crucial skill for many data science roles, and this course helps to understand and develop feature engineering. With a blended learning approach, Python for data science along with concepts like data wrangling, mathematical computing, and more can be learnt.					

Course Objectives		The objective of the course is to familiarize the learners with the concepts of **Applied Data Science** and attain **Employability** through E**xperiential Learning** techniques.
Course Out Comes		On successful completion of this course the students shall be able to: 1. Understand Numpy and Matrix Operations [Knowledge] 2. Analyze the need for data preprocessing and visualization techniques. [Comprehensive] 3. Demonstrate the performance of different supervised learning algorithms like decision Tree, Random Forest, Linear Regression, Logistic Regression etc. [Application] 4. Apply unsupervised learning algorithms like K-Means, K-Medoids etc for grouping the given data. [Applicaion]
Course Content:		

Module 1	**Introduction to Data Science, Python Data Structures, Python Numpy Package**	**Quiz**		**Knowledge based quiz**	**No. of sessions:8**
	Data Science - Need, Applications, Difference between data analysis and data analytics. Python-Variables, data types, control structures, Operators, Simple operations, Array and its operations, Numpy operations, Matrix and its operations				
Module 2	**Data preparation and preprocessing using Pandas dataframe, Exploratory Data Analysis, Data Visualization**	**Assignment**		**Data Visualization**	**No. of sessions:10**
	Dealing missing values, Normalization, statistical description about the data, Accessing the data, Summary of the data, Relationship between the data, Data Visualization using matplotlib				
Module 3	**Supervised Learning Algorithms**	**Design an algorithm using Example**		**Random Forest**	**No. of sessions:10**
	Decision Tree Algorithm, ID3 Classifier, Random Forest, Classifier Accuracy, Linear Prediction, Logistic Regression – Case study				
Module 4	**Unsupervised Learning Algorithms**	**Case Study**		**Conduct a case study on how data sets can be gathered and implemented in real time application.**	**No. of sessions:10**
	Various distance Function, Dissimilarity between the mixed types of data, K-Means Algorithm, K-Medoids Algorithm -Case Study				
	List of Laboratory Tasks: 1. Introduction to R tool for data analytics science 2. Basic Statistics and Visualization in R 3. K-means Clustering 4. Association Rules 5. Linear Regression 6. Logistic Regression 7. Naive Bayesian Classifier 8. Decision Trees 9. Simulate Principal component analysis 10. Simulate Singular Value Decomposition				

	Targeted Application & Tools that can be used: • IBM SPSS • Julia and Jupyter Notebook • Matplotlib
	Project work/Assignment:
	1. Design forest fire and wildfire prediction system. 2. Driver Drowsiness Detection System with OpenCV & Keras 3. Credit Card Fraud Detection using Python.
	Textbook(s): 1. Applied Data Science with Python and Jupyter-Alex Galea,Packt Publishing,October2018 2. Data Visualization in Python with Pandas and Matplotlib Paperback –DavidLandup, June 16, 2021
	References: 1.Data Science with Python and Dask- Jesse Daniel,1st Edition,July30,2019 **Weblinks:** • Udemy: https://www.udemy.com/course/applied-data-science-with-python-specialization-mhm/ • NPTEL online course : https://nptel.ac.in/courses/106106179 • https://presiuniv.knimbus.com/user#/home
	Topics relevant to "EMPLOYABILITY SKILLS": Data Science, Decision Tree Algorithm **for developing Employability Skills through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.**

Course Code: CSE3415	Course Title: Could Data Engineering Type of Course : Theory	L-T- P-C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	CSE3155					
Anti-requisites	nil					
Course Description	This Course is designed to introduce the concepts of Cloud Computing as a new computing paradigm. Cloud Computing has emerged in recent years as a new paradigm for hosting and delivering services over the Internet. The students can explore various Cloud Computing terminology, principles and applications. Understanding different views of the Cloud Computing such as theoretical, technical and commercial aspects. **Topics include:** Evolution of cloud computing and its services available today, Introduction, Architecture of cloud computing, Infrastructure, platform, software, Types of cloud, Business models, cloud services, Collaborating using cloud services, Virtualization for cloud, Security, Standards and Applications.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Could computing and Virtualization** and attain **Employability** through **Participative Learning** techniques.					
Course Out Comes	**On successful completion of the course the students shall be able to:** • Describe fundamentals of cloud computing, virtualization and cloud computing services. • Discuss high-throughput and data-intensive computing. • Explain security and standards in cloud computing. • Demonstrate the installation and configuration of virtual machine.					
Course Content:						
Module 1					**10 Sessions**	
	Introduction to Cloud and Virtualization Cloud Computing at a Glance, Historical Developments, Building Cloud Computing Environments, Computing Platforms and Technologies, Virtualization, Characteristics of Virtualized Environments Taxonomy of Virtualization Techniques, Virtualization and Cloud Computing, Technology Examples, Cloud Computing Architecture, IaaS, PaaS, SaaS, Types of Clouds, Economics of Cloud					
Module 2					**10 Sessions**	
	High Throughput and Data Intensive Computing: Task computing, MPI applications, Task based programming, Introduction to DIC, Technologies for DIC, Aneka Map Reduce Programming					
Module 3					**09 Sessions**	
	Cloud Security and Standards : Cloud Security Challenges, Software-as-a-Service Security, Application standards, Client standards, Infrastructure and Service standards.					
Module 4					**09 Sessions**	
	Cloud Platforms, Advances in cloud: introduction to **Amazon Web Services**: Introduction to **Google App Engine,** Introduction to **Microsoft Azure.** **Media Clouds - Security Clouds - Computing Clouds - Mobile Clouds – Federated Clouds – Hybrid Cloud**					
	Text Book 1. John Rittinghouse and James Ransome, "Cloud Computing, Implementation, Management and Security", CRC Press. 2. Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi, "Mastering Cloud Computing", McGraw Hill Education.					

	References 1. David E.Y. Sarna, "Implementing and Developing Cloud Applications", CRC Press. 2. Anthony T Velte, Toby J Velte, Robert Elsenpeter, "Cloud Computing: A Practical Approach", Tata McGraw-Hill. **Web resources:** https://presiuniv.knimbus.com/user#/home
	Topics relevant to "EMPLOYABILITY SKILLS": Aws, Azure, APIs, Aneka Cloud Platform, EC2, Installation of VM Workstation**,** Infrastructure Security Challenges for developing **Employability Skills** through **Participative Learning techniques**. This is attained through assessment component mentioned in course handout..

Course Code: CSE3416	**Course Title:** Federated Learning **Type of Course: Theory Only Course**	**L- T-P- C**	2	0	2	3

Version No.		1
Course Pre-requisites		CSE3155
Anti-requisites		
Course Description		Federated Learning can improve the performance of models by leveraging the diversity of the data across different devices. In this course, students will learn basics of Federated Learning and will be able to apply the real-time updates of the model in various practical scenarios.
Course Objective		The objective of the course is to get exposure to need for distributed model updates and understand the importance of privacy and security in machine learning techniques.
Course Out Comes		**On successful completion of the course the students shall be able to:** • Describe the key concepts and architecture of Federated Learning. (Knowledge) • Apply different methods to develop federated learning systems. (Comprehension) • Apply optimization techniques in Federated Learning (Application) • Construct and scale a simple federated system (Application) • Evaluate privacy and security concerns in Federated Learning and implement privacy-preserving techniques (Application)
Course Content:		

Module 1	**Introduction to Federated Learning:**	Assignment		Data Collection/Interpretation	**10 Sessions**
	Topics: Introduction to Federated Learning – Overview of Federated Learning: Definition, History, and Applications – Concepts and Terminology – Federated Learning Architecture -Machine Learning Perspective - Security & Privacy in Federated Learning – Federated Learning vs Centralized Learning: Comparison and Contrast.				
Module 2	**Horizontal and Vertical Federated Learning**	Case studies / Case let		Case studies / Case let	**13 Sessions**
	Topics: Horizontal Federated Learning (HFL) -Definition and Architecture of Horizontal Federated Learning – Federated Averaging (FedAvg) Algorithm – Improvements on the FedAvg Algorithm. Vertical Federated Learning (VFL) – Definition and Architecture of Vertical Federated Learning – VFL Algorithms: Secure Federated Linear Regression, Secure Federated Tree Boosting. Federated Learning with Non-IID Data – Heterogeneity in Federated Learning -Stratification and Local Updated Rules – Advanced Optimization Techniques in Federated Learning -Adaptive Learning Rate -Momentum and Weight Decay				
Module 3	**Federated Transfer Learning and Security**	Case studies / Case let		Case studies / Case let	**14 Sessions**
	Topics: Federated Transfer Learning (FTL) – Framework of Federated Transfer Learning – Homomorphic Encryption in FTL – FTL Training Process -FTL Prediction Process – Security Analysis of FTL – Secret Sharing based FTL Security in Federated Learning – Protecting Against Data Leakage in FL -Private Parameter Aggregation for FL – Data Leakage in FL Advanced Security Issues -Dealing with Byzantine Threats to Neural Networks in FL.				
	Targeted Application & Tools that can be used: Building and simulating federated learning systems using tensorFlowFederated (TFF), PySyft, Google Colab /				

	Jupyter Notebook. Simulating and deploying FL across nodes using Cloud Platforms (AWS, GCP, Azure) It can help small, medium and large businesses in any sector keep information assets secure.
	Project work/Assignment:
	Assignment: Practical Applications and Case Studies -Real-world Applications of Federated Learning
	Text Book **T1** Federated learning comprehensive overview of methods and applications Springer Nature Switzerland AG; 1st ed. 2022 edition By Heiko Ludwig (Editor), Nathalie Baracaldo **T2** Federated Learning (Synthesis Lectures on Artificial Intelligence and Machine Learning) , by Ronald J. Brachman, Francessa Rossi, and Peter Stone, Series Editors, Released 30 December 2019. Publisher(s): Morgan & Claypool Publishers. .
	References **R1 Federated Learning with Python by Kiyoshi Nakayama PhD, George Jeno, O'Reilly Media, Inc. Pub.** **R2 What-is-federated learning? By Emily Glanz, Nova Fallen, O'Reilly Media, Inc. Pub.** **E book link R1:** http://www.iso.org/iso/home/standards/management- standards/iso27001.html **E book link R2:** https://www.oreilly.com/library/view/what-is-federated/9781098107253/ch03.html
	Topics relevant to development of "SKILL DEVELOPMENT": Security Policy Implementation, Security Roles, for development of Skill Development through Participative Learning Techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE3417	Course Title: Edge Computing Type of Course: Theory Only Course Discipline Elective	L-T-P-C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	CSE3155					
Anti-requisites	**Nil**					
Course Description	In this course, we will study significant tools and applications that comprise today's cloud computing platform, with a special focus on using the cloud for big data applications. The course covers various topics such as the evolution of computing industry, cloud computing basics and edge computing. The course provides information on the different types of edge compute deployments, different types of edge compute services (such as CDN Edge, IOT Edge, and Multi-access Edge (MEC)). The course also educates the students on the different vendor platforms, software services, standard bodies and open source communities available for edge computing. Students will also create a research project of their choosing.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Edge Computing and attain **Employability** through **Problem Solving** Methodologies.					
Course Out Comes	On successful completion of the course the students shall be able to: CO1 Understand the principles, architectures of edge computing (Knowledge) CO2 Describe IoT Architecture and Core IoT Modules (Comprehension) CO3 Summarize edge to Cloud Protocols (Comprehension) CO4 Describe Edge computing with RaspberryPi (Comprehension)					
Course Content:						

Module 1	IoT and Edge Computing Definition and Use Cases	Term paper/Assignment/Case Study	Programming/Simulation/Data Collection/any other such associated activity		**9 Sessions**
	Topics: Introduction to Edge Computing Scenario's and Use cases - Edge computing purpose and definition, Edge computing use cases, Edge computing hardware architectures, Edge platforms, Edge vs Fog Computing, Communication Models - Edge, Fog and M2M.				
Module 2	IoT Architecture and Core IoT Modules	Term paper/Assignment/**Case Study**	Programming/Simulation/Data Collection/any other such associated activity		**9 Sessions**
	Topics: A connected ecosystem,IoT versus machine-to-machine versus, SCADA, The value of a network and Metcalfe's and Beckstrom's laws, IoT and edge architecture, Role of an architect, Understanding Implementations with examples-Example use case and deployment, Case study – Telemedicine palliative care, Requirements, Implementation, Use case retrospective.				
Module 3	RaspberryPi	Term paper/Assignment/Case Study	**Programming/Simulation**/Data Collection/any other such associated activity		**10 Sessions**
	Topics: Introduction to RaspberryPi, About the RaspberryPi Board: Hardware Layout and Pinouts, Operating Systems on RaspberryPi, Configuring RaspberryPi, Programming RaspberryPi, Connecting Raspberry Pi via SSH, Remote access tools, Interfacing DHT Sensor with Pi, Pi as Webserver, Pi Camera, Image & Video Processing using Pi.				
Module 4	Edge to Cloud Protocols	Term paper/Assignment/Case Study	Programming/Simulation/Data Collection/any other such associated activity		**7 Sessions**
	Topics: Implementation of Microcomputer RaspberryPi and device Interfacing, Edge to Cloud Protocols-				

	Protocols,MQTT, MQTT publish-subscribe, MQTT architecture details, MQTT state transitions,MQTT packet structure, MQTT data types, MQTT communication formats, MQTT 3.1.1 working example.			
Module 5	Edge computing with RaspberryPi	Term paper/Assignment/Case Study	**Programming/Simulation**/Data Collection/any other such associated activity	**7 Sessions**
	Topics: Edge computing with RaspberryPi, Industrial and Commercial IoT and Edge, Edge computing and solutions.			
	Targeted Application & Tools that can be used: • **Application** : Smart Surveillance Video Stream Processing at the Edge for Real-Time Human Objects Tracking. • **Tools** :Eclipse ioFog : An integrated development environment built by the Eclipse Foundation, backed by IBM. Eclipse ioFog is the organization's open-source edge computing platform.			
	Project work/Assignment: Mention the Type of Project /Assignment proposed for this course			
	Exploring topics such as developing scalable architectures, moving from closed systems to open systems, and ethical issues rising from data sensing, addresses both the challenges and opportunities of Edge computing presents. Students can harness federating Edge resources, middleware design issues, data management and predictive analysis, smart transportation and surveillance applications, and more. A coordinated and integrated solutions can be provided by thorough knowledge of the foundations, applications, and issues that are central to Edge computing.			
	Text Book 1. IoT and Edge Computing for Architects - Second Edition, by Perry Lea, Publisher: Packt Publishing, 2020, ISBN: 9781839214806 2. Raspberry Pi Cookbook, 3rd Edition, by Simon Monk, Publisher: O'Reilly Media, Inc., 2019, ISBN: 978149204322.			
	Topics relevant to "EMPLOYABILITY SKILLS": Implementation of Microcomputer RaspberryPi and device Interfacing for developing **Employability Skills** through **Problem Solving methodologies**. This is attained through assessment component mentioned in course handout.			

Course Code: CSE3418	Course Title: Network Security and Firewall Management Type of Course: Integrated	L-T- P- C	2	0	2	3
Version No.	1					
Course Pre-requisites	CSE3155					
Anti-requisites						
Course Description	This course provides an in-depth study of various network attacks techniques and methods to defend against them. A number of threats and vulnerabilities of the Internet will be covered, including various vulnerabilities of TCP/IP protocols, denial of service (DOS), attacks on routing, attacks on DNS servers, TCP session hijacking, and so on. This course will also cover defending mechanisms, including intrusion detection, firewalls, tracing the source of attacks, anonymous communication, IPsec, virtual private network, and PKI. To make it easy for students to understand these attacks, basics of the TCP/IP protocols will also be covered in the course.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Firewall and Internet security** and attain **Skill Development** through **Problem Solving** Methodologies.					
Course Out Comes	**On successful completion of the course the students shall be able to:** • To identify elements of firewall design, types of security threats and responses to security attacks. • Examine security incident postmortem reporting and ongoing network security activities. • Construct code for authentication algorithms. • Develop a signature scheme using Digital signature standard. • Demonstrate the network security system using open source tools					
Course Content:						

Module 1	Introduction to Firewall	Assignment		Data Collection/Interpretation	**12 Sessions**
	Introduction of Firewall in computer network,Categories of firewall,How firewall works,Types of firewall, Firewall location and Configuration,Firewall Policies,Firewall Biasing,Network Architecture,Net masks,Packet filters,Stateful firewalls,Resources				
Module 2	**Computer security**	Case studies / Case let		Case studies / Case let	**12 Sessions**
	Topics: Attacks on Computers and Computer Security: Need for Security, Security Approaches, Principles of Security Types of Attacks. Transport Level Security: Web Security Considerations, Secure Sockets Layer, Transport Layer Security, HTTPS, Secure Shell (SSH)				
Module 3	**Network Security**	Quiz		Case studies / Case let	**10 Sessions**
	Topics: Overview of Network Security:Elements of Network Security , Classification of Network Attacks ,Security Methods ,Symmetric-Key Cryptography :Data Encryption Standard (DES),Advanced Encryption Standard (AES) , Public-Key Cryptography :RSA Algorithm ,Diffie-Hellman Key-Exchange Protocol , Authentication :Hash Function , Secure Hash Algorithm (SHA) , Digital Signatures.				
Module 4	Cyber laws and Compliance Standards	Quiz		Case studies / Case let	**11 Sessions**
	Topics: Kerberos:Working ,ASS,TGS,SS-Internet security protocols-AH,ESP,Models-Transport and tunnel-Email security,Public key Infrasturcture,Certificates,certificates authority.Cyber Crime: Introduction,Hacking,Digital forgery,Cyber Stalking,Identify theft and Fraud,Cyber terrorism,Cyber defamation,Crime against individual,Government,Property.				
	List of Laboratory Tasks:				

	1. Perform encryption, decryption using the following substitution techniques (i) Ceaser cipher, (ii) playfair cipher iii) Hill Cipher iv) Vigenere cipher 2. Perform encryption and decryption using following transposition techniques i) Rail fence ii) row & Column Transformation 3. Apply DES algorithm for practical applications. 4. Apply AES algorithm for practical applications. 5. Implement RSA Algorithm using HTML and JavaScript 6. Implement the Diffie-Hellman Key Exchange algorithm for a given problem. 7. Calculate the message digest of a text using the SHA-1 algorithm. 8. Implement the SIGNATURE SCHEME – Digital Signature Standard. 9. Demonstrate intrusion detection system (ids) using any tool eg. Snort or any other s/w. 10. Automated Attack and Penetration Tools Exploring N-Stalker, a Vulnerability Assessment Tool 11. Defeating Malware i) Building Trojans ii) Rootkit Hunter
	Targeted Application & Tools that can be used
	Text Book **T1 :** Behrouz A Forouzan, Data and Communications and Networking, Fifth Edition, McGraw Hill, Indian Edition **T2:** James F Kurose and Keith W Ross, Computer Networking, A Top-Down Approach, Sixth edition, Pearson,2017
	References **R1:** Andrew S Tanenbaum, Computer Networks, fifth edition, Pearson Edition **R2**: Nader F Mir, Computer and Communication Networks, 2nd Edition, Pearson, 2014. **Web resources:** 1. https://networklessons.com/cisco/asa-firewall 2. https://www.udemy.com/course/cisco-asa-firewall-lab-guide 3. https://geekflare.com/learn-network-security
•	• **Topics relevant to development of "Skill Development": AES, Network Security for Skill Development through Problem Solving methodologies. This is attained through assessment component mentioned in course handout.**

Course Code: CSE3419	**Course Title:** Information Security and Management **Type of Course: Theory Only Course**		**L- T-P- C**	3	0	0	3

Version No.		1
Course Pre-requisites		CSE3155
Anti-requisites		
Course Description		The course explores information security through some introductory material and helps gain an appreciation of the scope and context of information security. It includes a brief introduction to cryptography, security management, network and computer security. It allows a student to begin a fascinating journey into the study of information security and develop an appreciation of some key security concepts. The course concludes with a discussion of a simple model of the information security in industry and explores skills, knowledge and roles required for employability. A student will be able to determine and analyze potential career opportunities in this profession.
Course Objective		The objective of the course is to familiarize the learners with the concepts of Information Security and Management and attain **Employability** through **Participative Learning** techniques.
Course Out Comes		**On successful completion of the course the students shall be able to:** • Describe the basic concept of information security. (Knowledge) • Explain the concepts and methods of cryptography. (Comprehension) • Demonstrate the aspects of risk management. (Application)
Course Content:		

Module 1	**Information Security Management:**	Assignment		Data Collection/Interpretation	**10 Sessions**
	Topics: Information Security Overview, Threat and Attack Vectors, Types of Attacks, Common Vulnerabilities and Exposure (CVE), Security Attacks, Fundamentals of Information Security, Computer Security Concerns, Information Security Measures.				
Module 2	**Fundamentals of Information Security and Data Leakage**	Case studies / Case let		Case studies / Case let	**13 Sessions**
	Topics: Key Elements of Networks, Logical Elements of Networks, Critical Information Characteristics, Information States. What is Data Leakage and Statistics, Data Leakage Threats, Reducing the Risk of Data Loss, Key Performance Indicators (KPI), Database Security.				
Module 3	**Information Security Policies and Management**	Case studies / Case let		Case studies / Case let	**14 Sessions**
	Topics: Information Security Policies-Necessity-Key Elements and Characteristics, Security Policy Implementation, Configuration, Security Standards-Guidelines and Frameworks, Security Roles and Responsibilities, Accountability, Roles and Responsibilities of Information Security Management, Team Responding to Emergency Situation- Risk Analysis Process.				
	Targeted Application & Tools that can be used: An ISMS is a systematic approach to managing sensitive company information so that it remains secure. It includes people, processes and IT systems by applying a risk management process. It can help small, medium and large businesses in any sector keep information assets secure. The ISO 27000 family of standards helps organizations keep information assets secure.				

	Using this family of standards will help your organization manage the security of assets such as financial information, intellectual property, employee details or information entrusted to you by third parties. ISO/IEC 27001 is the best-known standard in the family providing requirements for an information security management system (ISMS).
	Project work/Assignment:
	Assignment:
	Text Book **T1** Management of Information Security by Michael E.Whilman and Herbert J.Mattord **T2** Information Security: The Complete Reference, Second Edition, 2nd Edition. by Mark Rhodes-Ousley. Released April 2013. Publisher(s): McGraw-Hill. .
	References **R1 Title, Cryptography & Network Security (Sie) 2E. Author, Forouzan. Publisher, McGraw-Hill Education (India) Pvt Limited.** **R2 Information Systems Security, 2ed: Security Management, Metrics, Frameworks and Best Practices. Nina Godbole.** **E book link R1:** http://www.iso.org/iso/home/standards/management- standards/iso27001.html **E book link R2:** http://csrc.nist.gov/publications/nistpubs/800-55-Rev1/SP800-55-rev1.pdf 3LINKS: pu.informatics.global , https://sm-nitk.vlabs.ac.in.
	Topics relevant to development of "SKILL DEVELOPMENT": Security Policy Implementation, Security Roles, for development of Skill Development through Participative Learning Techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE3420	**Course Title: Network Intrusion Detection and Prevention** **Type of Course:1] Program Core** **2] Theory Only**		**L- T-P- C**	3	0	3

Version No.		1.0
Course Pre-requisites		CSE3155
Anti-requisites		**NIL**
Course Description		Objective of the course is to Understand when, where, how, and why to apply Intrusion Detection tools and techniques in order to improve the security posture of an enterprise. Apply knowledge of the fundamentals and history of Intrusion Detection in order to avoid common pitfalls in the creation and evaluation of new Intrusion Detection Systems and Analyze intrusion detection alerts and logs to distinguish attack types from false alarms.
Course Objectives		The objective of the course is to familiarize the learners with the concepts of **Intrusion Detection and Prevention System** and attain **Skill Development** through **Participative Learning** techniques.
Course Out Comes		On successful completion of the course the students shall be able to: • Understand about the intruders. • Define intrusion detection and prevention policies • Explain the fundamental concepts of Network Protocol Analysis and demonstrate the skill to capture and analyze network packets. • Use various protocol analyzers and Network Intrusion Detection Systems as security tools to detect network attacks and troubleshoot network problems.
Course Content:		

Module 1	**Introduction to Intrusion Detection and Prevention System**	Assignment	Programming Task	**10 Sessions**

Topics
Understanding Intrusion Detection – Intrusion detection and prevention basics – IDS and IPS analysis schemes, Attacks, Detection approaches –Misuse detection – anomaly detection – specification based detection – hybrid detection. Internal and external threats to data, Need and types of IDS, Information sources,Host based information sources, Network based information sources.

Assignment: Demonstrating the skills to capture and analyze network packets using network packet analyzer.

Module 2	**Intrusion Prevention System**	Assignment	Programming Task	**10 Sessions**

Topics:
Intrusion Prevention Systems, Network IDs protocol based IDs, Hybrid IDs, Analysis schemes, thinking about intrusion. A model for intrusion analysis, techniques, Responses, requirement of responses, Types of responses, mapping responses to policy Vulnerability analysis, credential analysis, non-credential analysis. Architecture models of IDs and IPs.

Assignment: Applying Intrusion detection in security applications.

Module 3	**Applications and tools**	Assignment	Programming/Data analysis task	**12 Sessions**

Topics:
Tool Selection and Acquisition Process – Bro Intrusion Detection – Prelude Intrusion Detection – Cisco Security IDS – Snorts Intrusion Detection – NFR security. Introduction to Snort, Snort Installation Scenarios, Installing Snort, Running Snort on Multiple Network Interfaces, Snort Command Line Options. Step-By-Step Procedure to Compile and Install Snort Location of Snort Files, Snort Modes Snort Alert Modes

Assignment: Demonstrate the working with Snort Rules, Rule Headers, Rule Options and The Snort Configuration File.

Module 4	Legal issues and organizations standards	Assignment	Programming/Data analysis task	**9 Sessions**

Law Enforcement / Criminal Prosecutions – Standard of Due Care – Evidentiary Issues, Organizations and Standardizations.

Assignment: Addressing common legal concerns and myths about Intrusion Detection system

Textbooks

T1. Carl Endorf, Eugene Schultz and Jim Mellander " Intrusion Detection & Prevention", 1st Edition, Tata McGraw-Hill, 2004.
T2. Earl Carter, Jonathan Hogue, "Intrusion Prevention Fundamentals", Pearson Education, 2006.

References

R1. Rafeeq Rehman : " Intrusion Detection with SNORT, Apache, MySQL, PHP and ACID," 1st Edition, Prentice Hall , 2003.
R2. Christopher Kruegel, Fredrik Valeur, Giovanni Vigna: "Intrusion Detection and Correlation Challenges and Solutions", 1st Edition, Springer, 2005.
R3. Paul E. Proctor, "The Practical Intrusion Detection Handbook ",Prentice Hall , 2001.

Weblinks:
https://www.youtube.com/watch?v=RYB4cG8G2xo
https://www.coursera.org/lecture/detecting-cyber-attacks/intrusion-detection-systems-UeDqJ

Topics relevant to "SKILL DEVELOPMENT": Agent development for intrusion detection for Skill Development through **Participative Learning techniques**. This is attained through assessment component mentioned in course handout.

Course Code: CSE3421	Course Title: Principles and Practices of Web Security Type of Course: Integrated	L- T-P- C	2	0	2	3
Version No.	1					
Course Pre-requisites	CSE3155					
Anti-requisites	NIl					
Course Description	The purpose of this course this course is to introduce you to the field of web security by understanding web functionality and various security validations. The web is our gateway to many critical services and is quickly evolving as a platform to connect all our devices. Web vulnerabilities are growing on a year-to-year basis and designing secure web applications is challenging. The course covers fundamental concepts of web security principles, web vulnerability and exploitation, various attacks on web applications, and a few basic topics on web encryption.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Web Security** and attain **Skill Development** through **Experiential Learning** techniques.					
Course Out Comes	**On successful completion of the course the students shall be able to:** • Define the fundamentals of web applications and validation [Knowledge] • Recognize the significance of password and authentication in web applications[Comprehension] • Explain the importance of session management in web [Comprehension] • Apply web attack techniques to find vulnerabilities in web applications [Application]					
Course Content:						

Module 1	**Introduction**	Quiz		Comprehension based Quiz on web fundamentals	**10 Sessions**
	Topics: Web Functionality, Encoding Schemes, Mapping the Application - Enumerating the Content and Functionality, Analyzing the Application Bypassing, Client-Side Controls: Transmitting Data Via the Client, Capturing User Data, Handling Client-Side Data Securely - Input Validation, Blacklist Validation - Whitelist Validation - The Defense in-Depth Approach - Attack Surface Reduction, Rules of Thumb, Classifying and Prioritizing Threats.				
Module 2	**Web Application Authentication**	Assignment		Comprehensive based assignment on Web authentication	**11 Sessions**
	Topics: Authentication Fundamentals- Two Factor and Three Factor Authentication, Web Application Authentication- Password Based, Built-in, HTTP, Single Sign-on, Custom Authentication, Validating credentials - Secured Password Based Authentication: Attacks against Password, Importance of Password Complexity - Design Flaws in Authentication Mechanisms - Implementation Flaws in Authentication Mechanisms - Securing Authentication.				
Module 3	**Session Management &Web Security Principles**	Quiz		Comprehension based Quiz on web security techniques.	**11 Sessions**
	Topics: Need for Session Management, Weaknesses in Session Token Generation, Weaknesses in Session Token Handling, Securing Session Management; Access Control: Access Control Overview, Common Vulnerabilities, Attacking Access Controls, Securing Access Control. Origin Policy, Exceptions, Browser security Principles- Cross Site Scripting and Cross Site Request Forgery, File Security Principles: Source Code Security, Forceful Browsing, Directory Traversals.				
Module 4	**Web Application**	Assignment		Comprehension based assignment on web	**10 Sessions**

	Vulnerability			vulnerabilities	

Topics:
Attacking data-stores and backend components- Injecting into Interpreted Contexts, injecting into SQL, NoSQL, XPath, LDAP, Injecting OS Commands, Manipulating File Paths, Injecting into XML Interpreters, Injecting into Back-end HTTP Requests, Injecting into Mail Services, Attacking application logic-real world logic flaws, Attacking users-Cross site scripting-varieties of XSS,XSS attacks in action, finding and exploiting XSS vulnerabilities, preventing XSS attacks, Other techniques-cookie based Attacks, HTTP Header Injection

List of Laboratory Tasks:

Task 01: Practical knowledge of known vulnerabilities in CGI, LAMP stacks, REST APIs cross-site scripting
Task 02: HTTP and setting up stacks, the various types of databases Access Controls, Vulnerabilities
Task 03: SQL injection and prevention
Task 04: Study of web authoring tools
Task 05: Testing web applications
Task 06: Cross site request forgery attack lab
Task 07: Web tracking

Targeted Application & Tools that can be used

1. Wordpress tool can be used for building websites with possible vulnerabilities.
2. Tools such as Nmap and Nessus can be used for web attack demonstration**.**

Project work/Assignment:

Assignment:
Group assignment to identify and write different web exploits to demonstrate vulnerabilities in web applications.

Text Book
T1Dafydd Stuttard, Marcus Pinto, "The Web Application Hacker's Handbook", Willey Publishing Inc.

References
R1 B. Sullivan, V. Liu, and M. Howard, "Web Application Security", A B Guide. New York: McGraw-HillEducation, 2011**R2**Web Application Security: Exploitation and Countermeasure for Modern Web Applications, by AndrewHoffman
E book link R1: https://presiuniv.knimbus.com/user#/home
E book link R2 : https://presiuniv.knimbus.com/user#/home

Web resources:
NPTEL / Swayam Link: Introduction to Information Security I, IIT Madras
https://nptel.ac.in/courses/106106129
PU Library Link :https://puniversity.informaticsglobal.com/login

Topics relevant to "EMPLOYABILITY SKILLS":
Session Management &Web Security Principles and Web Application vulnerability for **Skill Development** through **Experiential Learning Techniques.** This is attained through the assessment component mentioned in the course handout.

Course Code: CSE3422	**Course Title: Penetration Testing and Risk Assessment** **Type of Course: Theory Only Course**	**L-T- P- C**	3	0	0	3
Version No.	1.0					
Course Pre-requisites	CSE3155					
Anti-requisites	**NIL**					
Course Description	This course explores the tools that can be used to perform information gathering. This course also covers how vulnerability can be carried out by means of tools or manual investigation, and analysis of common attacks in data, mobile applications and wireless networks.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Vulnerability Assessment and Penetration Testing** and attain **Employability** through **Problem Solving** Methodologies.					
Course Out Comes	**On successful completion of the course the students shall be able to:** • Understand the basic principles for information gathering and detecting vulnerabilities in the system. • Determine the security threats and vulnerabilities in SDN networks and web applications. • Able to use the exploits in mobile applications and wireless networks • Understand the metasploit and metrepreter are used to automate the attacks and penetration testing techniques.					
Course Content:						
Module 1	**Information Gathering, Host Discovery and Evading Techniques**	Assignment		Theory		**9 Sessions**
	Topics: Introduction - Terminologies - Categories of Penetration Testing - Phases of Penetration Test -Penetration Testing Reports - Information Gathering Techniques - Active, Passive and Sources of Information Gathering – Approaches, Host discovery - Scanning for open ports and services- Types of Port, Vulnerability Scanner Function, pros and cons - Vulnerability Assessment with NMAP - Testing, SCADA environment with NMAP					
Module 2	**Vulnerability Scanner in SDN Networks and Web application**	Quiz		Theory		**10 Sessions**
	Topics: Nessus Vulnerability Scanner - Safe check – Silent dependencies - Port Range Vulnerability Data Resources, SDN Data plane, Control Plane, Application Plane. SDN security attack vectors and SDN Harderning, Authentication Bypass with Insecure Cookie Handling - XSS Vulnerability - File inclusion vulnerability - Remote file Inclusion -Patching file Inclusions - Testing a website for SSI Injection.					
Module 3	**Mobile Application Security and wireless network Vulnerability analysis**	Quiz		Theory		**11 Sessions**
	Topics: Types of Mobile Application Key challenges in Mobile Application and Mobile application penetration testing methodology, Android and ios Vulnerabilities - OWASP mobile security risk - Exploiting WM - BlackBerry Vulnerabilities - Vulnerability Landscape for Symbian - Exploit Prevention -Handheld Exploitation, WLAN and its inherent insecurities Bypassing WLAN Authentication uncovering hidden SSIDs MAC Filters Bypassing open and shard authentication - Advanced WLAN Attacks Wireless eavesdropping using MITM session hijacking over wireless – WLAN Penetration Test Methodology.					
Module 4	**Exploits**	Quiz		Theory		**8 Sessions**
	Topics:					

	Architecture and Environment- Leveraging Metasploit on Penetration Tests, Understanding - Metasploit Channels, Metasploit Framework and Advanced Environment configurations – Understanding the Soft Architecture, Configuration and Locking, Advanced payloads and add on modules Global datastore, module datastore, saved environment Meterpreter.
	Targeted Application & Tools that can be used: This course helps the students to understand the threats and vulnerabilities using NMAP.
	Project work/Assignment:
	Project Assignment:
	Text Book 1. Rafay Baloch, Ethical Hacking and Penetration Testing Guide, CRC Press, 2015. ISBN : 78-1-4822-3161-8. 2. Dr. Patrick Engebretson, The Basics of Hacking and Penetration Testing Ethical Hacking and Penetration Testing made easy , Syngress publications, Elsevier, 2013. ISBN :978-0-12-411644-3. 3. Mayor, K.K.Mookey, Jacopo Cervini, Fairuzan Roslan, Kevin Beaver, Metasploit Toolkit for Penetration Testing, Exploit Development and Vulnerability Research, Syngress publications, Elsevier, 2007. ISBN : 978-1-59749-074-0
	References 1. Mastering Modern Web Penetration Testing By Prakhar Prasad,October 2016 PacktPublishing. 2. SQL Injection Attacks and Defense 1st Edition, by Justin Clarke-Salt, Syngress Publication **Web resources:**https://onlinecourses.nptel.ac.in/noc19_cs68/preview - **IIT Kharagpur,** Prof. Indranil Sen Gupta
	Topics relevant to development of "EMPLOYABILITY SKILLS": Exploitation, Penetration testing techniques, for development of Employability skills through the Participative Learning Techniques. This is attained through the assessment components mentioned in course handout.

Course Code: CSE3423	Course Title: Go Programming Type of Course: Theory Only Course	L- T-P- C	3	0	0	3
Version No.	1.0					
Course Pre-requisites	CSE1004 Problem Solving using C					
Anti-requisites	NIL					
Course Description	Go is an open source programming language created by Google. Go is expressive, concise, clean, and efficient. Its concurrency mechanisms make it easy to write programs that get the most out of multicore and networked machines. Go compiles quickly to machine code yet has the convenience of garbage collection and the power of run-time reflection. It's a fast, statically typed, compiled language that feels like a dynamically typed, interpreted language. It is gaining popularity and it is continuing to grow rapidly in industries such as Dropbox, Uber etc. This course will provide an introduction to the Go programming essentials to students of Engineering through lecture hours with demonstrations. Topics: Topics covered in this course are go program structure; data types and control statements; Composite Types – arrays, slices, strings, runes, bytes, hash maps; functions; methods; garbage collection essentials – pointers, structs, interfaces; error handling; Concurrency – go routines and channels, Packages – import and create custom packages and applications of Go					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **GO Programming** and attain Employability Skills through Problem Solving techniques.					
Course Out Comes	**On successful completion of the course the students shall be able to:** CO1: Identify primitive programming constructs in GO. (**Knowledge**) CO2: Discuss composite data types with concepts of modular programming. (**Comprehension**) CO3: Implement garbage collection using pointers, structs, interfaces and modules. (**Application**) CO4: Apply concurrent programming and test routines with applications. **(Application)**					
Course Content:						

Module 1	**Introduction to Go Programming Language**	Assignment	Data Collection/Interpretation		**10 Sessions**
Topics: **[Knowledge]** Feature of Go language, Installing and Configuring the development environment- Go tools and playground. Structure of Go program; Basic types-numbers, boolean, strings, runes. Variables- declaration, zero values, naming, rules, conversions, constants, multiple variables. Introduction to packages, functions from other packages, println, reading input, Control Structures - if, switch, for, programming exercises using control statements.					
Module 2	**Composite types and functions**	Assignment	Data Collection/Interpretation		**9 Sessions**
Topics: **[Comprehension]** Composite types - arrays, slices, slices with overlapping storage, Structs. Functions-declaring, parameters, returning multiple values, variadic functions; Programming exercises					
Module 3	**Pointers, Structs, Interfaces and modules**	Quiz	Case studies / Case let		**9 Sessions**
Topics: **[Application]** Pointers: *and & operator, types, pointers with functions, garbage collector – history, Methods and Interfaces, Modules,packages – importing and creating custom packages; Programming exercises.					

Module 4	**Concurrency and Applications**	Quiz	Case studies / Case let		**7 Sessions**

Topics: **[Application]**

Concurrency using Go routines, multiple go routines, channels – channel operations, Testing- writing test, Go test command, Core Packages for – strings, containers and lists, Writing Web Applications, Basic Statistical Computations, histogram plotting, encryption and decryption.

Targeted Application & Tools that can be used:

1. https://go.dev/play/
2. https://go.dev/doc/install

Project work/Assignment:

Text Book

T1 1. John Badner,"Learning Go: An Idiomatic Approach to Real World Go Programming", Oreilly, California,2021.

References

R1. 1. Alan A.A. Donovan and Brian W. Kernighan, "The Go Programming Language", Pearson Education, India,2016.

R2. Tsoukalos M. Mastering Go: Create Golang production applications using network libraries, concurrency, machine learning, and advanced data structures. Packt Publishing Ltd; 2019 Aug 29.

Web resources: https://www.golangprograms.com/go-language.html

EBSCO database of Presidency University:https://puniversity.informaticsglobal.com/login

W3**.** GO document: https://go.dev/doc/

Online tool for program execution:

- GO Play Ground - https://go.dev/play/
- Download and install: https://go.dev/doc/install

Topics relevant to development of "Employability": Go Programming basics for developing **Employability Skills** through Problem Solving methodologies. This is attained through assessment component mentioned in course handout.

Course Code: CSE3424	**Course Title:Advanced DBMS** **Type of Course: Core** **Theory &Integrated Laboratory**	**L-T-P-C**	2	0	2	3
Version No.	**1.0**					
Course Pre-requisites	**Database Management System (CSE3156)**					
Anti-requisites	**NIL**					
Course Description	**The purpose of this course is to make the students revisit RDBMS transactions first. Then introduce them with Distributed, Parallel, and NoSQL database concepts. They include the main characteristics, advantages, and disadvantages of each one of them. Importance and differences among them are noted. Need to transit from RBMS to NoSQL is discussed. The striking features of distributed, parallel and NoSQL are considered and studied.** **The associated laboratory provides a chance to have hands-on concepts learned during this course.**					
Course Objective	**This course is designed to improve the learners' <u>EMPLOYABILITY SKILLS</u> by learning the working on Database using MySQL.**					
Course Outcomes	**On successful completion of this course the students shall be able to:** (1) Recall the transactions in RDMS **(2)** Explain advanced features of distributed, parallel, and NoSQL databases. **(3)** Illustrate the features in Distributed database **(4)** Employ Parallel database concepts in real life applications.					
Course Content:						
Module 1	**Transactions in RDBMS**	Quiz	Comprehension based Quizzes and assignments.	**06Classes**		
Topics: RDBMS -Transaction control state diagram, ACID properties of transaction, Schedules in transactions - Serial, Non-Serial and Serializable, Serializability-Conflict and View, Conflict Serializability check by Precedency Graph, Concurrency Control – Lock Based and Time Stamp Based.						
Module 2	**NoSQL Databases**	Programming and Mini Project	Laboratory experiments and Mini Projects on NoSQL Topics using MongoDB/ Casandra.	**06Classes**		
Topics: NoSQL Introduction – Scale Out, Commodity Hardware, Brief History, Features – Non-Relational, Schema Free, Simple API, and Distributed. NoSQL Architectures/Data Models - Document, Columnar, Key-Value, and Graph. Transaction in NoSQL- BASE for reliable database transactions, Achieving Horizontal Scalability with Database Sharding, CAP theorem. Case Study: MongoDB/Casandra/ AWS/ HBase						
Module 3	**Distributed Databases**	Assignment	Assignment on main topics of Distributed Databases	**06Classes**		
Topics: Loosely Coupled, Characteristics of Distributed Databases, Local and Global view of applications, Distributed Processing, Types – Homogeneous and Heterogeneous, Distributed Data Storage – Replication and Fragmentation, Fragmentation – Horizontal and Vertical Type, Difference between Centralized and Distributed Databases.						
Module 4	Parallel Databases	Assignment	Assignment on main topics of Parallel Databases	**06 Classes**		
Topics: Tightly Coupled, Features of parallel databases, Shared Memory, Shared Disk, Shared Nothing Systems. Advantages of each of these schemes, Advantages and Disadvantages of Parallel Databases, Differences between Parallel and Distributed Databases.						
Install MONGODB						

https://www.javatpoint.com/mongodb-create-database
Create any one of the following databases.
Employee, Student, University, Banking, or Online Shopping
Drop database
Create Collection: In MongoDB db.createCollection(name,option) is used to create collection.
Drop Collection

List of Laboratory Tasks:(7 X 2= 14 Sessions)

Level 1: Perform CRUD operations (Insert, Update, Delete and Query Documents) on 'Student' Database.
Level 2: Do MongoDB text search on 'Employee' Database.

Experiment No. 2: Try experiments on MongoDB Operators
Level 1: Perform queries involving MongoDB Query and Projection Operators using 'Student' Database.
Level 2: Do queries involving MongoDB update operator on 'Employee' Database.

Experiment No. 3:Explore different query modifiers.
Level 1: Perform different query modifiers on 'Student' Database.
Level 2: Try various query modifiers on 'Employee' Database.

Experiment No. 4:Explore Aggregation commands.
Level 1: Implement different aggregation commands on 'Student' Database.
Level2: Perform various aggregation commands on 'Employee' Database.

Experiment No. 5:Explore Authentication commands.
Level 1: Try authentication commands on 'Student' Database.
Level 2: NA

Experiment No. 6:Explore Replication Commands
Level 1: Try all replication commands on 'Student' Database.
Level2: Implement replication commands on 'Employee' Database.

Experiment No.7:Try Sharding Commands.
Level1: Explore Sharding Commands on 'Student' Database.
Level 2: Implement Sharding Commands on 'Employee' Database.

Targeted Application & Tools that can be used:
MongoDB is to be installed and used.

Project work/Assignment:

Each batch of students (self-selected batch mates) will identify projects, such as, Library, Banking, and Reservation etc.,and do it. Concepts of NoSQL, like, CRUD operations, supporting ad hoc queries, indexing flexibility, assisting replication, creating capped collections, and Retrieving data from multiple documents.
Sample Mini Projects:
1. Content Management System
Clubbing the content assets like text and HTML into a single database helps provide a better user experience. MongoDB has an excellent toolset not only for storing and indexing but also for controlling the structure of a content management system. You can easily design a web-based CMS by using the model proposed by "Metadata and Asset Management" in MongoDB. Additionally, you can use "Storing Comments" to model user comments on blog posts.
2. Gaming Project
Data is an essential part of making video games work. Some typical examples of gaming data include player profiles, matchmaking, telemetry, and leaderboards.
The common thread between all games is that they all have a specific goal. And you have to achieve multiple objectives or pay your way out to reach the end goal. This may involve steps like watering your plants, growing vegetables, serving food in a restaurant, and so on.

Textbook(s):

1. **Sadalage, P. & Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, 1st Edition, 2019(Wiley Publications).**
2. **Stefano Ceri, Giuseppe Pelagatti , Distributed Databases: Principles and Systems,, 2017(McGraw Hill Education).**

References

1. **Elmasri R and Navathe S B, "Fundamentals of Database System",7th Edition, 2017(Pearson Publication).**
2. **Pivert. *NoSQL Data Models: Trends and Challenges*, 1st edition(Wiley).**

Topics related to development of "FOUNDATION":Transaction, CRUD Operations, Replication, and Sharding
Topics related to development of "EMPLOYABILITY": Project implementations in software, batch wise presentations
Topics related to development of "HUMAN VALUES AND PROFESSIONAL ETHICS": Team Dynamics during Mini Project Development.

rse Code: 3425	**rse Title:** Programming in C# and .NET **e of Course:** Theory Integrated Lab	**L-T-P-C**	1	0	4	3
Version No.	**1.0**					
rse Pre-requisites	**CSE 1006 : Problem Solving using Java**					
i-requisites						
rse Description	.NET is a software framework which is designed and developed by Microsoft. It is used to develop Form-based applications, Web-based applications, and Web services. There is a variety of programming languages available on the .Net platform, VB.Net, C# etc. C# is one of the most popular languages of .NET framework. It is used to build applications for Windows, phone, web etc. It provides a lot of functionalities and also supports industry standards. This course intends to provide the basic concepts of .NET framework and various components of the .NET framework architecture. This course also provides the features of C# programming language to design and implement console, desktop-based applications and web based applications.					
Course Objective	**This course is designed to improve the learners' <u>EMPLOYABILITY SKILLS</u> by learning the working on Database using MySQL.**					
Course Outcomes	On successful completion of this course the students shall be able to: 1. Understand the fundamental concepts and benefits of .NET framework and its components.[Comprehension] 2. Illustrate the Object-Oriented paradigm using C# Language.[Knowledge] 3. Develop different types of applications by applying the C# programming concepts and database connectivity.[Application] 4. Demonstrate the use of event handling mechanism[Application]					
Course Content:						
lule 1	cepts of .NET Technology, Implementation of .NET, Components of .NET framework		nonstration of Assembly, oduction to IDE		**+ Pr 4+ 1(6)**	
ics: erstand the motivation behind the .NET platform, Common Language Infrastructure (CLI). Know the role of the Common Type System (CTS), the Common Language Specification (CLS) and the Common Language Runtime (CLR), Understand the assembly, metadata, namespace, type distinction, Contrast single-file and multi-file assemblies, Know the role of the Common Intermediate Language (CIL),Introduction to .NET Core						
lule 2	Programming constructs, OOPS concepts	Based Assignments,	ds on Session for the Concepts, Creating a Console application		**3+6(15)**	
Topics: Program structures, Types and Variables, Expressions, Statements, Structs, Class, Objects Name Spaces - Constructor and Destructors, Function Overloading & Inheritance, Operator Overloading, Array, Interfaces, Modifiers -, Working with Console input & Output, Property and Indexers , Enum ,Interfaces						
lule 3	nt handling, Data base Connectivity	based Assignments	tice Exercises using Visual Studio.NET		**(9)**	
ics: gates and events, Exception handling, Advantage of ADO.NET, ADO.NET Architecture, .NET Providers, Connection, Command, Data Set, Data Reader. Working with Connection Oriented and Connection less approach, Programming Window Forms Applications: The notifies - subscribers paradigm for handling ts. .NET framework for handling GUI events						
dule 4	P.NET	based Assignments	ctice Exercises using Visual Studio.NET		**2(6)**	
pics:						

roduction to Web Forms, Basic working of WebForms, Introduction to Web Forms, Connectivity with the database , validation Controls

of Practical Tasks:

eriment 1:[Module 1]

Level 1: Demonstrate the .NET framework and Visual studio IDE for writing C# code.

Level 2: Demonstrate the .NET framework and inside of Assembly using ildasm tool.

eriment 2: [Module 2]

Level 1: University wants to gift for those date of birth falls on February 29th . Create a C# program that will accept the employee's birth year. Check the leap year and issue them with surprise gift.

Level 2: A developer wants to check the given input is in Fibonacci series or not.

eriment 3: [Module 2]

Level 1 : A teacher is asked to create mark list of her class students. The class consists of 10 students and they have 5 different subjects. Store the student's name and five subject marks also. Calculate the total of all subject marks and display them.

Level 2: A class teacher is storing the students 'name and Roll number. Write a program to help to sort out the roll number using different sorting techniques.

eriment 4: [Module 2]

Level 1: Design a class to represent a bank account. Include the following members: Data Members: - Name of the depositor, Account Number, Type of Account, Balance amount in the account and methods : To assign initial values, To deposit an amount, To withdraw an amount after checking balance, To display name and the balance. Write a C# program to demonstrate the working of the various class members.

Level 2: Define a class 'Person' with data members name and age. Also include following: Default Constructor and parameterized constructor, Input method which takes values from user and assigns to data members, Output method to display all data . Create 5 objects of 'Person' class using array of objects and call all the methods of a class.

eriment 5: [Module 2]

el 1: Write a C# program to show single and multilevel inheritance.

vel 2: Create a class 'Emp' by extending Person class with additional data member empno, position with following features:

Default constructor

Parameterized constructor

Input method which takes values from user and assigns to data members and calls input method of Person

Output method to display all data and calls output method of Person

ne a class Manager by extending Emp with data member bonus. Provide necessary constructors and override input and output methods. Create objects of manager in main.

eriment 6: [Module 2]

el 1: Calculate the area of different shapes using method overloading.

el 2: Class teacher created different groups in a class and store the data in that. In order to make common announcements and activities, teacher merged all data into a single group. Write a code to merge two groups into one.

eriment 7: [Module 2]

el 1: Class Teacher stores students marks in an array. Teacher is searching for highest and lowest marks of the class and number of students scored those marks. Write a program to help teacher to do the same.

el 2: Create an application for currency converter.

Experiment 8: [Module 3]

el 1: EC is updating their database of new voters. If the user's age is less than 18, application should raise the exception.

el 2: Develop a desktop based application for displaying employees salary and leave balance.

eriment 9: [Module 3]

Level 1: Create a login screen and prompts for the user name and Password. If the user exists in the database , show welcome message to the user when the button is clicked.

Level 2: Company wants to create a calculator application using C#.

eriment 10: [Module 3]

Level 1: University stores the CSE students' data in the database and display the student details whenever required. Administrator allows to insert, update and modify the data. Implement this.

Level 2: University decides to conduct online quiz for CSE students. Create a windows-based application to implement this.

eriment 11: [Module 4]

Level 1: University is organizing a cultural festival and organizing teams wants to collect registration for various events with the help of web page. Design a registration form for collecting the participant details.

el 2: University is decided to display all information about the various departments in their website.

gn a web site to show the above mentioned.

eriment 12: [Module 4]

vel 1: XYZ corporation wants to review their product. So company is creating a feedback form and validating the data with the help of validation controls. Design a feedback form with validation controls.

el 2: XYZ corporation wants to review their product. So company is creating a feedback form and validating the data with the help of validation controls. Design a feedback form with validation controls and display the comparison chart of various months.

Targeted Application & Tools that can be used:
Microsoft Visual studio.NET 2022, Visual Studio Code.

Project work/Assignment:
Problem Solving: Design of Algorithms and implementation of programs.
Programming: Implementation of given scenario using .NET.

tbook(s):

1. **Herbert Schildt , "C# 4.0 The Complete Reference", Fourth Edition, TMH**
2. **Matthew Macdonald,"ASP.NET: The Complete Reference", McGraw Hill Education**

erences:

1. Joseph Albahari and Ben Albhari, "C# 3.0/4.0 in NUTSHELL", O'REILLY.
2. Andrew Troelsen, "C# and the .NET Platform" 1st edition Apress
3. Matthew Macdonald,"Beginning ASP.NET 4.5 in C#", Wiley India

Online References

1.C# Tutorial (C Sharp) (w3schools.com)
2. https://docs.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/tutorials/
3.https://docs.microsoft.com/en-us/aspnet/tutorials

Topics relevant to development of "Employability": Web Application developing Employability Skills through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE3426	Course Title: Front-end Full Stack Development	L- T-P- C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	CSE1006 – Problem Solving Using Java					
Anti-requisites	**NIL**					
Course Description	This intermediate course enables students to perform front-end full stack development, with emphasis on employability skills. The course covers key technologies and architectures that enables the student to design and implement front-end. On successful completion of this course, the student shall be able to pursue a career in full-stack development. The students shall develop strong problem-solving skills as part of this course.					
Course Objectives	The objective of the course is to familiarize the learners with the concepts Front end Full Stack Development and attain Employability through experiential Learning techniques.					
Course Outcomes	**On successful completion of the course the students shall be able to:** **Describe the fundamentals of DevOps and Front-end full stack development. [Comprehension]** **Illustrate a basic web design using HTML, CSS, Javascript. [Application]** **Illustrate development of a responsive web. [Application]** **Apply concepts of Angular.js to develop a web front-end. [Application]**					
Course Content:						

Module 1	Fundamentals of DevOps	Project		Programming	**04 Sessions**
	Topics: Introduction to Agile Methodology; Scrum Fundamentals; Scrum Roles, Artifacts and Rituals; DevOps – Architecture, Lifecycle, Workflow & Principles; DevOps Tools Overview – Jenkins, Docker, Kubernetes. Review of GIT source control.				
Module 2	Web Design & Development	Project		Programming	**03 Sessions**
	Topics: HTML5 – Syntax, Attributes, Events, Web Forms 2.0, Web Storage, Canvas, Web Sockets; CSS3 – Colors, Gradients, Text, Transform; **Assignment:** Develop a website for managing HR policies of a department.				
Module 3	Responsive web design	Project		Programming	**08 Sessions**
	Topics: BootStrap for Responsive Web Design; JavaScript – Core syntax, HTML DOM, objects, classes, Async; Ajax and jQuery Introduction **Assignment:** Design and develop a website that can actively keep track of entry-exit information of a housing society.				
Module 4	Fundamentals of Angular.js	Project		Programming	**15 Sessions**
	Topics: Setting up Development & Build Environment: Node.js and NPM; Introduction to TypeScript; Working with OOP concepts with TypeScript; Angular Fundamentals; Angular CLI; Introduction to TypeScript; Debugging Angular applications; Components & Databinding in Depth; Angular Directives; Using Services & Dependency Injection; Angular Routing; Observables; Handling Forms in Angular Apps; Output transformation using Pipes; Making Http Requests; Authentication & Route Protection; Dynamic Components; Angular Modules & Optimizing Angular Apps; Deploying an Angular App; Angular Animations; Adding Offline Capabilities with Service Workers; Unit Testing in Angular Apps (Jasmine, Karma). Overview of React.js **Assignment:** Develop a software tool to do inventory management in a warehouse.				
	Targeted Application & Tools that can be used: **Application Area is to Design and Analyzing the efficiency of Algorithms. This fundamental course is used by all application developers.**				

	Professionally Used Software: GCC compiler.
	Text Book: Fender, Young, "*Front-end Fundamentals",* Leanpub, 2015 Northwood, Chris, "*The Full Stack Developer: Your Essential Guide to the Everyday Skills Expected of a Modern Full Stack Web Developer",* APress, 2018
	References: R1. Flanagan D S, *"Javascript : The Definitive Guide"* 7th Edition. 7th ed. O'Reilly Media; 2020. Alex Libby, Gaurav Gupta, and Asoj Talesra. *"Responsive Web Design with HTML5 and CSS3 Essentials",* Packt Publishing, 2016 Duckett J Ruppert G Moore J. *"Javascript & Jquery : Interactive Front-End Web Development.";* Wiley; 2014. Web Reference: /www.youtube.com/watch?v=JGNTYXkVCVY&list=PLd3UqWTnYXOkTSBCBNyyhxo_jxlY_uTWA&index=2 Web Reference: https://www.freecodecamp.org/news/frontend-web-developer-bootcamp/ https://puniversity.informaticsglobal.com:2229/login.aspx?direct=true&db=nlebk&AN=2233842&site=ehost-live https://nptel.ac.in/courses/106102064
	Topics relevant to development of "Employability": DevOps Tools Overview – Jenkins, Docker, Kubernetes for developing Employability Skills through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE3427	Course Title: Java Full Stack Development	L- T-P- C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	**CSE1006 Problem Solving Using Java**					
Anti-requisites						
Course Description	This advanced level course enables students to perform full stack development using Java, with emphasis on employability skills. The key technologies used for Full Stack development is based on either Java technology or .NET technology. In this course, the focus is on using Java, and the related technologies/tools like Java EE, Java Persistence, Hibernate, Maven, Spring Core, etc. On successful completion of this course, the student shall be able to pursue a career in full-stack development. The students shall develop strong problem-solving skills as part of this course.					
Course Objectives	This course is designed to improve the learners' EMPLOYABILITY SKILLS by using PROBLEM SOLVING Methodologies.					
Course Outcomes	**On successful completion of the course the students shall be able to:** **1] Practice the use of Java for full stack development [Application]** **2] Show web applications using Java EE. [Application]** **3] Solve simple applications using Java Persistence and Hibernate [Application]** **4] Apply concepts of Spring to develop a Full Stack application. [Application]** **5] Employ automation tools like Maven, Selenium for Full Stack development. [Application]**					
Course Content:						

Module 1	Introduction	Project		Programming	**03 Sessions**
	Topics: Review of Java; Advanced concepts of Java; Java generics; Java IO; New Features of Java. Unit Testing tools.				
Module 2	Java EE Web Applications	Project		Programming	**05 Sessions**
	Topics: Introduction to Eclipse & Tomcat; JSP Fundamentals; Reading HTML form Data with JSP; State Management with JSP; JSP Standard Tag Library - Core & Function Tags; Servlet API Fundamentals; ServletContext, Session, Cookies; Request Redirection Techniques; Building MVC App with Servlets & JSP; Complete App - Integrating JDBC with MVC App **Assignment:** Develop an application for managing HR policies of a department.				
Module 3	Java Persistence using JPA and Hibernate	Project		Programming	**06 Sessions**
	Topics: Fundamentals of Java Persistence with Hibernate; JPA for Object/Relational Mapping, Querying, Caching, Performance and Concurrency; First & Second Level Caching, Batch Fetching, Optimistic Locking & Versioning; Entity Relationships, Inheritance Mapping & Polymorphic Queries; Querying database using JPQL and Criteria API (JPA) **Assignment:** Design and develop a website that can actively keep track of entry-exit information of a housing society.				
Module 4	Spring Core	Project		Programming	**10 Sessions**
	Topics: Spring Core, Spring MVC, Spring Boot REST API; Understanding Spring Framework; Using Spring MVC; Building a Database Web App with Spring and Hibernate o Spring AOP (Aspect Oriented Programming); Implementing Spring Security; Developing Spring REST API; Using Spring Boot for Rapid Development **Assignment:** Develop a software tool to do inventory management in a warehouse.				

Module 5	Automation tools	Project		Programming	**06 Sessions**

Topics:
Introduction to Automation Tools; Apache Maven: Maven Fundamentals, Software Setup - Commandline and Eclipse, pom.xml and Directory Structure, Multi-Module Project Creation, Scopes, Dependency Management, Profiles; Functional/BDD Testing using Selenium, Selenium Fundamentals and IDE, Selenium WebDriver, Installation and Configuration, Locating WebElements, Driver Commands, WebElement Commands
Assignment: Illustrate the use of automation tools in the development of a small software project.

Targeted Application & Tools that can be used:

Application Area is to Design and Analyzing the efficiency of Algorithms. This fundamental course is used by all application developers.

Professionally Used Software: Eclipse, NetBeans, Hibernate, Selenium, Maven, GIT.

Project work/Assignment:

Problem Solving: Design of Algorithms and implementation of programs.
Programming: Implementation of given scenario using Java.

Text Book:

T1. Fender, Young, "*Front-end Fundamentals",* Leanpub, 2015

References

R1. Soni, Ravi Kant. *"Full Stack AngularJS for Java Developers: Build a Full-Featured Web Application from Scratch Using AngularJS with Spring RESTful."* , Apress, 2017.
R2. Mardan, Azat. *"Full Stack JavaScript: Learn Backbone.js, Node.js and MongoDB."*, Apress, 2015

Course Code: CSE3428	Course Title: .NET Full Stack Development			L- T-P- C	2	0	2	3
Version No.		1.0						
Course Pre-requisites		CSE1006 – Problem solving using java						
Anti-requisites		**CSE3427 Java Full Stack Development**						
Course Description		This advanced level course enables students to perform full stack development using .NET, with emphasis on employability skills. The key technologies used for Full Stack development is based on either Java technology or .NET technology. In this course, the focus is on using .NET and the related technologies/tools like C#, ASP.NET, Entity Framework Core, etc. On successful completion of this course, the student shall be able to pursue a career in full-stack development. The students shall develop strong problem-solving skills as part of this course.						
Course Objectives		This course is designed to improve the learners' EMPLOYABILITY SKILLS by using PROBLEM SOLVING Methodologies.						
Course Outcomes		**On successful completion of the course the students shall be able to:** **1] Practice the use of C# for developing a small application [Application]** **2] Show web applications using Entity Framework. [Application]** **3]Solve simple web applications that use SQL and ASP.NET [Application]** **4] Apply concepts of ASP.NET to develop a Full Stack application. [Application]**						
Course Content:								

Module 1	C# Programming for Full Stack Development	Project		Programming	**10 Sessions**
	Topics: .NET Framework Fundamentals, Visual Studio IDE Fundamentals, C# Language Features, Working with arrays and collections, Working with variables, operators, and expressions, Decision and iteration statements, Managing program flow and events, Working with classes and methods, OOP concepts, Properties, Auto Implemented, Delegates, Anonymous Methods and Anonymous Types, Extension methods, Sealed Classes/Methods, Partial Classes/Methods, Asynchronous programming and threading, Data validation and working with data collections including LINQ, Handling errors and exceptions, Working with Files, Unit Testing – Nunit framework **Assignment:** Develop a small application for managing library using C#.				
Module 2	Entity Framework Core 2.0	Project		Programming	**06 Sessions**
	Topics: Entity Framework Core 2.0 Code First Approach; Introduction To Entity Framework and EDM; Querying the EDM; Working With Stored Procedures; Advanced Entity Framework - DbContext [EF6]; Advanced Operations; Performance Optimization; Data Access with ADO.NET **Assignment:** Develop an application for managing HR policies of a department.				
Module 3	ASP.NET	Project		Programming	**06 Sessions**
	Topics: ASP.NET Core, ASP.Net Core 3.1 MVC, ASP.NET Core Middleware and Request pipeline, Review of SQL using MS SQL, Working With Data In Asp.Net, Razor View Engine, State Management In Asp. Net MVC & Layouts; **Assignment:** Develop a web application to mark entry/exit of guests in a building.				
Module 4	ASP.NET	Project		Programming	**08 Sessions**
	Topics: Introduction To Models, Validations In Asp.Net MVC, Authentication and Authorization In Asp.Net MVC,				

	Advanced Asp. Net MVC - Ajax Action Link In MVC, Advanced Asp.Net MVC - Ajax Forms In MVC, Microsoft Testing Framework – Unit Testing the .NET Application **Assignment:** Develop a software tool to do inventory management in a warehouse.
	Targeted Application & Tools that can be used: **Application Area is to Design and Analyzing the efficiency of Algorithms. This fundamental course is used by all application developers.** **Professionally Used Software: Visual Studio**
	Project work/Assignment:
	Problem Solving: Design of Algorithms and implementation of programs. **Programming: Implementation of given scenario using .NET.**
	Text Book: T1. Fender, Young, "*Front-end Fundamentals",* Leanpub, 2015 T2. Valerio De Sanctis, *"ASP.NET Core 5 and Angular: Full-stack web development with .NET 5 and Angular 11"*, 4th Edition, Packt, 2021.
	References R1. Benjamin Perkins, Jon D. Reid, *"Beginning C# and .NET"*, Wiley, 2021 Reid, 2021. R2. Piotr Gankiewicz, *"Full Stack .NET Web Development"*, Packt Publishing, 2017. R3. Tamir Dresher, Amir Zuker, Shay Friedman, *"Hands-On Full-Stack Web Development with ASP.NET Core"*, Packt Publishing, 2018. R4. Dustin Metzgar, *"Exploring .NET core with microservices, ASP.NET core, and Entity Framework Core"*, Manning, 2017.

Course Code: CSE3429	**Course Rust Programming** **Type of Course: Core** **Theory &Integrated Laboratory**	**L-T-P-C**	2	0	2	3
Version No.	**1.0**					
Course Pre-requisites	**CSE1004 Problem Solving using C**					
Anti-requisites	**NIL**					
Course Description	Rust is for students and those who are interested in learning about systems concepts. Using Rust, many people have learned about topics like operating systems development. The Rust programming language helps students write faster, more reliable software. High-level ergonomics and low-level control are often at odds in programming language design; Rust challenges that conflict. Through balancing powerful technical capacity and a great developer experience, Rust gives developers the option to control low-level details such as memory usage without all the hassle traditionally associated with such control. The associated laboratory provides a chance to have hands-on concepts learned during this course.					
Course Objective	**This course is designed to improve the learners' EMPLOYABILITY SKILLS by learning the Rust Programming language.**					
Course Outcomes	**Upon the successful completion of this course students will be able to:** • **Create a Rust project, including proper Cargo configuration.** • **Translate a design into a working Rust program.** • **Explain and remedy type and lifetime errors encountered during Rust programming.** • **Use structs, enums and traits as intended in the construction of Rust programs.** • **Apply references, boxes, cells and reference counting in Rust programming.** • **Divide a Rust crate into multiple source files using the module system.** • **Write tests and documentation using the Rust infrastructure.**					
Course Content:						

Module 1	Introduction to Rust Programming	Quiz		**10 sessions**
Topics: Introduction to Rust Programming: Reasons to adopt Rust –Use cases – Opportunities- Language Features- Advantages- Installation- First Example-Rust Data types- Variables – Constants-String-Operators- Branching and Looping				
Module 2	Features of Rust	Programming and Mini Project		**12 sessions**
Topics: Unique Features of Rust: Tuple- Common types (Option, Result)- Array-Ownership – Borrowing- Slices Structures- Enums-Collections (vector, string, hashmap, iterator) - Modules - Rust Collections-Error Handling- Input Output Generics- Package Manager-Iterator- Closure-Smart Pointers – Concurrency				
Module 3	**Ownership**	Assignment		**11 sessions**
Topics: Ownership + move semantics - Borrowing (References) – Lifetimes - Smart pointers (Box, Rc/Arc, Cell/Refcell) - Modularising projects - Documenting code - Unit testing - Documentation testing				
Module 4	Generics and Concurrency	Assignment		**12 sessions**
Topics: Generics (and monomorphization) – Traits - Static vs dynamic dispatch – Closures - Function types (fn, FnOnce, FnMut, Fn) – Macros - Fearless concurrency – Threads - Sync primitives (Mutex, RwLock, mpsc, etc.) - Current state and future of Rust				

Install Rustup
https://www.rust-lang.org/tools/install
Experiment No. 1
Create and run a new project using the conventions of Cargo
Create a project to Covert Fahrenheit to Celsius.
Experiment No. 2
Create a project to display binary equivalent of an integer, perform AND, and shift operations.
Create a dollar pattern
Create a guessing game program that will ask for user input, process that input, and check that the input is in the expected form.
Experiment No. 3
Create a project to display the magnitude and angle of the given complex numbers.

Create a project with functions to check a number is prime or not, count primes, add primes and display primes within the given range.

Experiment No. 4
Create a simple programs using simple struct, associated functions and Enums
Implement a Turtle builder struct to allow building a Turtle object. Perform rotation with angle, move forward and backward.
Experiment No. 5
Write a program to solve the quadratic equation using command line arguments.
Write a program on Word histogram on accepting a file, count the occurrences using HashMap, sort and display.
Experiment No. 6
Create a program to simulate John Conway's Game of Life using array and thread sleep.

Create a project to make changes to Turtle type so that it is placed in a library module and main function uses the module.
Experiment No. 7
Write a program to handle error that main returns a Result type, making the necessary code changes.
Create a new library project named generics to implement stack and queue operations.

Targeted Application & Tools that can be used:
Rust is to be installed and used.
https://www.rust-lang.org

Project work/Assignment:

Problem Solving: Design of Algorithms and implementation of programs.
Programming: Implementation of given scenario using Rust.

Textbook(s):

1. **Klabnik, Steve, and Carol Nichols. *The Rust programming language*. No Starch Press, 2023. Publisher: William Pollock**

References

1. **Jim Blandey, Jason Orendorff and Leonora F.S. Tindall, "Programming Rust – Fast, safe system Development",2nd Edition, 2021(O'Reilly Publication).**

https://rustbook.cs.brown.edu

Topics related to development of "FOUNDATION":Features of Rust
Topics related to development of "EMPLOYABILITY": Project implementations in software, batch wise presentations
Topics related to development of "HUMAN VALUES AND PROFESSIONAL ETHICS": Team Dynamics during Mini Project Development.

Course Code: CSE3430	**Course Title: Introduction to Fintech** **Type of Course: Program Core Theory & Lab Integrated**	**L-T-P-C**	2	0	2	3

Version No.		1.0
Course Pre-requisites	•	NIL
Anti-requisites		NIL
Course Description		This course aims to familiarize students with the FinTech ecosystem and the disruptive and innovative forces of emerging technology within the finance sector. A core component of a specialized business administration program, it delivers essential, leading-edge knowledge in financial technology, crucial for professionals entering the banking and financial services industry.
Course Object		The objective of the course is to familiarize the learners with the concepts of Introduction to Fintech attain **Skill Development** through **Experiential Learning** techniques.
Course Out Comes		On successful completion of the course the students shall be able to: **CO1:** Describe the historical development of financial technology. (Understand) **CO2:** Analyze the impact of financial technology on the financial services landscape. (Apply) **CO3:** Explain the fundamental technical aspects of financial technology. (Apply) **CO4:** Identify and interpret key technological trends within the financial services sector. (Apply)
Course Content:		

Module 1	Introduction to FinTech	Assignment				**14 Sessions**
	Topics: What is FinTech Industry? Evolution of FinTech, FinTech Evolution 1.0: Infrastructure, FinTech Evolution 2.0: Banking industry, FinTech Evolution 3.0 & 3.5: Startups and Emerging Markets, Importance of FinTech, Global FinTech Investment, Main FinTech Hubs					
Module 2	FinTech Reshaping Financial Services Industry-I	Assignment				**15 Sessions**
	Topics: FinTech in Payment Industry-Multichannel digital wallets, applications supporting wallets, onboarding and KYC application, FinTech in Lending Industry- Formal lending, Informal lending, P2P lending, POS lending, Online lending, Payday lending, Microfinance, Crowdfunding					
Module 3	FinTech as	Assignment				**16**

	disruptor empowering Financial Services Industry-II				**Sessions**
	Topics: FinTech in Wealth Management Industry-Financial Advice, Automated investing, Socially responsible investing, Fractional Investing, Social Investing. FinTech in Insurance Industry- P2P insurance, On-Demand Insurance, On-Demand Consultation, Customer engagement through Quote to sell, policy servicing, Claims Management, Investment linked health insurance.				
Module 4	Technology Disruptions enabling FinTech Innovations	Assignment			**15 Sessions**
	Topics: 4G and 5G networks fuelling FinTech Opportunities, transforming customer experience using Mobile Applications and smart phones, embedded sensors and social media, Cloud computing, Web 2.0, Rapid Web Design, JavaScript Technologies, IoT, Big Data, analytics and AI and Blockchain Case Studies: PayTm, Aadhar				
	Project work/Assignment:				
1.	2. **Assignment 1 on (Module 1 and Module 2)** 3. **Assignment 2 on (Module 3 and Module 4)**				
	Text Book 1) Parag Y Arjunwadkar (2018), FinTech: The Technology Driving Disruption in the financial service industry CRC Press. 2) Sanjay Phadke (2020), Fintech Future : The Digital DNA of Finance Paperback .Sage Publications 3) Pranay Gupta, T. Mandy Tham (2018). Fintech: The New DNA of Financial Services Paperback 4) RBI(2017). Report of working group on FinTech and Digital Banking				
	References **1. Bitcoin for Non-Mathematicians: Exploring the foundations of Crypto, SlavaGomzin/ Universal Publishers, USA, Latest 1 ST Edition 2020** **2. The Robotics Process Automation, Handbook: A Guide to Implementing, Tom Taulli/ Apress, Latest 1 ST Edition 2020** **Web Resources** W1. https://www.ibm.com/industries/banking-financial-markets/resources/omnichannelbanking-paper/ W2. https://thefinancialbrand.com/111080/evolution-future-digital-banking-baastransformation/				
	Topics relevant to development of "Employability": Real time Analysis of FinTech applications and opportunities. **Topics relevant to "PROFESSIONAL ETHICS":** Case studies on Paytm, AAdhar				

Catalogue prepared by		
Recommended by the Board of Studies on		
Date of Approval by the Academic Council		

urse Code: **E3431**	**urse Title: Banking Technology** **pe of Course: Program Core** **eory & Lab Integrated**	**T-P-C**	2	0	2	3

rsion No.		
urse Pre-requisites	•	NIL
ti-requisites		L
urse Description		This course explores the evolution and application of technology in banking, starting from branch computerization to centralized banking. It covers delivery channels such as ATMs, internet/mobile banking, and UPI/BHIM. Topics include risk management, treasury, forex operations, and CRM. Students will understand INFINET, SFMS, RTGS, NEFT, and e-payment systems. Emphasis is placed on practical implementation and emerging digital banking trends.
urse Object		The objective of the course is to familiarize the learners with the concepts of Banking Technology attain **Skill Development** through **Experiential Learning** techniques.
urse Out Comes		On successful completion of the course the students shall be able to: **CO1:** Understand the evolution of banking technology and its impact on modern banking operations.(Understand) **CO2:** Explain the role of delivery channels like ATMs, mobile banking, and UPI in digital banking systems.(Apply) **CO3: Evaluate** the effectiveness of risk, treasury, and data center management in banking operations.(Apply) **CO4: Analyze** the functioning of centralized banking and payment systems like RTGS, NEFT, and SFMS (Apply) **CO5:** Understand blockchain, cryptocurrency concepts, and analyze recent core banking software. (Understand)
urse Content:		

dule 1	Branch Operation and Core Banking	signment			**14 Sessions**

	Topics: Introduction and Evolution of Bank Management Analysis of Rangarajan - Committee Reports - Technological Impact in Banking Operations– Total Branch Computerization - Concept of Opportunities– Centralized Banking – Concept, Opportunities, Challenges & Implementation.				
odule 2	livery Channels	signment			**Sessions**
	Topics: Overview of delivery channels – Automated Teller Machine (ATM) – Phone Banking –Call centers – Internet Banking – Mobile Banking- USSD, UPI, BHIM – Payment Gateways – Card technologies – MICR electronic clearing.				
odule 3	ck office Operations	signment			**Sessions**
	Topics: Bank back office management –Inter branch reconciliation – Treasury Management– Forex Operations – Risk Management – Data center Management – Network Management – Knowledge Management (MIS/DSS/EIS) – Customer Relationships Management (CRM).				
odule 4	erbank Payment System	signment			**Sessions**
	pics: FINET Interface with Payment system Network– Structured Financial Messaging system –Electronic Fund transfer – RTGSS – Negotiated Dealing Systems & Securities Settlement Systems – Electronic Money – E Cheques.				
odule 5	ntemporary Issues in Banking				
	chniques Block Chain and Bit-coin – Crypto currency Analysis of Recent Core Banking Software-Case study.				
	oject work/Assignment:				
9.	**10. Assignment 1 on (Module 1 and Module 2)** 11. **Assignment 2 on (Module 3,4 and Module 5)**				
	Text Book 8) Financial Services Information Systems-Jessica Keyes Auerbach publication, October 2019. 9) **Rajesh, R.** (2020). *Banking Technology.* New Delhi: McGraw Hill Education. 10) IIBF X Taxmann's International Trade Finance – Complete | Expert-vetted Guide on—Trade Theories | ICC Rules | Regulatory Frameworks | Risk Management | Digitisation | LIBOR-ARR Transitions, February 2025 11) **Kalakota, R., & Robinson, M.** (2017). *E-Banking Management: Issues, Solutions, and Strategies.* New Delhi: Pearson Education.				
	References **1. Vasudeva,E–Banking, Common Wealth Publishers, New Delhi, 2010** **2. Turban Rainer Potter, Information Technology, John Wiely& Sons Inc,2012.** **3. Banking Technology – Indian Institute of Bankers Publication,2010.** **Web Resources** W8. https://www.ibm.com/industries/banking-financial-markets/resources/omnichannelbanking-paper/ W9. https://thefinancialbrand.com/111080/evolution-future-digital-banking-baastransformation/				
	Design and Develop the following Banking Software using the appropriate technologies:				

	■ **Mobile Banking** **▪ Balance Enquiry ▪ Cheque book Request ▪ Stop Cheque ▪ Credit/Debit Notification ▪ Bill Payment** ■ **Internet Banking** **▪ Electronic Funds Transfer ▪ Account Management ▪ Loan Application ▪ Registering of new bank services ▪ Customer Information Management** ■**ATM system** **▪ Balance Enquiry ▪ Withdrawal ▪ Deposit ▪ Pin change ▪ Mini statement**
	Topics relevant to development of "Employability": Real time Data Analysis for Banking Technology. **Topics relevant to "PROFESSIONAL ETHICS":** Mobile, Internet Banking for Project Development.

Catalogue prepared by		
Recommended by the Board of Studies on		
Date of Approval by the Academic Council		

Course Code: CSE3432	**Course Title: Blockchain Technology** **Type of Course: Theory**	**L-T-P-C**	2	0	2	3

Version No.		1.0
Course Pre-requisites	•	NIL
Anti-requisites		NIL
Course Description		This course provides a comprehensive introduction to the fundamental concepts and applications of blockchain technology. Students will explore the principles of decentralization, understand the mechanics of Bitcoin, delve into the world of smart contracts and alternative cryptocurrencies, and gain practical experience in developing and deploying smart contracts using industry-standard tools. The course emphasizes both the theoretical underpinnings and the practical implementation of blockchain solutions.
Course Object		The objective of the course is to familiarize the learners with the concepts of Blockchain Technology attain **Skill Development** through **Experiential Learning** techniques.

Course Out Comes		On successful completion of the course the students shall be able to: **CO1: Understand** blockchain and decentralization principles. (Understand) **CO2: Analyze** Bitcoin's operation and transactions. (Analyze) **CO3: Apply** smart contract concepts and explore altcoins. (Apply) **CO4: Utilize** Truffle for smart contract development and deployment. (Apply)				
Course Content:						
Module 1	**Fundamentals of Blockchain and Decentralization**	Assignment			**14 Sessions**	
	Topics: Introduction to Blockchain: the growth of blockchain technology, the history of blockchain and Bitcoin, distributed systems, blockchain, consensus. Decentralization: decentralization using blockchain, methods of decentralization, routes to decentralization, blockchain and full ecosystem decentralization, pertinent terminology, platforms for decentralization, innovative trends					
Module 2	**Bitcoin: The First Cryptocurrency**	Assignment			**15 Sessions**	
	Topics: Introduction to Bitcoin: bitcoin — an overview, cryptographic keys, transactions, blockchain, mining. The Bitcoin Network and Payments: the bitcoin network, wallets, bitcoin payments, innovation in bitcoin, advanced protocols, bitcoin investment and buying and selling Bitcoin. Bitcoin Clients and APIs: bitcoin client installation, experimenting further with bitcoin-cli, bitcoin programming					
Module 3	**Beyond Bitcoin: Altcoins and Smart Contracts**	Assignment			**16 Sessions**	
	Topics: Consensus Algorithms: introducing the consensus problem, analysis and design, classification, algorithms, choosing an algorithm. Alternative Coins: introducing altcoins, theoretical foundations, difficulty adjustment and retargeting algorithms, bitcoin limitations, extended protocols on top of bitcoin, development of altcoins, Initial Coin Offerings (ICOs). Smart Contracts: history, definition, ricardian contracts, smart contract templates, oracles, deploying smart contracts, the DAO					
Module 4	**Ethereum and the Decntralized Web (Web3)**	Assignment			**15 Sessions**	
	Topics: Ethereum 101: ethereum – an overview, the ethereum network, components of the ethereum ecosystem, Ethereum Virtual Machine (EVM), smart contracts, ethereum development environment . Further Ethereum: blocks and blockchain, wallets and client software, nodes and miners, APIs, tools, and DApps, supporting protocols, programming languages. Introducing Web3: contract deployment, exploring Web3 with Geth					

	Project work/Assignment:
1.	**2. Assignment 1 on (Module 1 and Module 2)** **3. Assignment 2 on (Module 3 and Module 4)**
	Text Book 1. Banafa, A. (2024). *Blockchain technology and applications*. River Publishers. 2. Ramachandran, M. (2025). *Blockchain engineering: Secure, sustainable frameworks for healthcare applications*. Springer. 3. Tanwar, S. (2022). *Blockchain technology: From theory to practice*. Springer. 4. Vyas, S., Shukla, V. K., Gupta, S., & Prasad, A. (Eds.). (2022). *Blockchain technology: Exploring opportunities, challenges, and applications*. CRC Press.
	References 1. Chuen, D. L. K. (Ed.). (2024). *Handbook of digital currency: Bitcoin, innovation, financial instruments and big data* (2nd ed.). Academic Press. 2. Idrees, S. M. & Nowostawski, M. (Eds.). (2023). *Transformations through blockchain technology: The new digital revolution*. Springer. 3. Jena, A. K., Panda, S. K., & Swain, S. K. (Eds.). (2022). *Blockchain technology: Applications and challenges* (Vol. 203). Springer. 4. Maleh, Y., Zhang, J., & Hansali, A. (2024). *Advances in emerging financial technology and digital money*. Routledge. 5. Rahman, H. (Ed.). (2025). *Blockchain technology applications in knowledge management*. IGI Global. **Web Resources** 1. Blockgeeks. Retrieved from https://www.google.com/search?q=blockgeeks.com 2. Bitcoin.org. Retrieved from https://bitcoin.org/ 3. CoinDesk. Retrieved from https://www.coindesk.com/ 4. Ethereum.org. Retrieved from https://ethereum.org/ 5. Investopedia. Retrieved from https://www.investopedia.com/ 6. Medium. Retrieved from https://medium.com/ 7. Solidity Documentation. Retrieved from https://docs.soliditylang.org/ 8. Truffle Suite Documentation. Retrieved from https://trufflesuite.com/docs 9. Web3.js Documentation. Retrieved from https://web3js.readthedocs.io/ 10. GitHub. Retrieved from https://github.com/ **YouTube Channels:** 1. Andreas Antonopoulos. Retrieved from https://www.youtube.com/@aantonop 2. Chainlink. Retrieved from https://www.youtube.com/@chainlink 3. Coin Bureau. Retrieved from https://www.youtube.com/@CoinBureau 4. Eat The Blocks. Retrieved from https://www.youtube.com/@EatTheBlocks 5. freeCodeCamp.org. Retrieved from https://www.youtube.com/@freecodecamp 6. MetaMask. Retrieved from https://www.youtube.com/@MetaMask 7. Patrick Collins. Retrieved from https://www.youtube.com/@patrickdcollins 8. Simply Explained. Retrieved from https://www.youtube.com/@SimplyExplained 9. The Defiant. Retrieved from https://www.youtube.com/@TheDefiant 10. Whiteboard Crypto. Retrieved from https://www.youtube.com/@WhiteboardCrypto

Experiment 1: Integrated Development Environments (IDEs) for Smart Contracts

- **Level 1:** Explore the features and interface of Remix IDE. Deploy a simple "Hello World" smart contract on the in-browser JavaScript VM. Observe the transaction details and contract interaction options.
- **Level 2:** Install and configure MetaMask browser extension. Connect MetaMask to the Remix IDE. Deploy the same "Hello World" contract to the Ganache private network via MetaMask. Examine the transaction process in both Remix and MetaMask.

Experiment 2: MetaMask in a Private Network

- **Level 1:** Set up a local Ganache private network. Add a custom network in MetaMask, configuring the RPC URL and Chain ID to connect to your Ganache instance. Create a new account in MetaMask and observe its balance.
- **Level 2:** Deploy a simple token contract (e.g., ERC-20 minimal) using Remix IDE and MetaMask on your private Ganache network. Transfer some tokens between the accounts you created in MetaMask and observe the balance changes.

Experiment 3: Smart Contract with Solidity - Basic Data Types and Structures

- **Level 1:** Write a Solidity smart contract that declares and initializes variables of different basic data types (uint, string, bool, address). Implement functions to read and modify these variables. Deploy and interact with the contract in Remix IDE.
- **Level 2:** Create a Solidity smart contract that utilizes structs and arrays. Implement functions to add, retrieve, and update elements within these data structures. Deploy and test the contract with various inputs in Remix IDE.

Experiment 4: Smart Contract with Solidity - Control Flow and Functions

- **Level 1:** Write a Solidity smart contract that uses `if-else` statements and `for` loops within its functions. Implement a function that performs a simple calculation based on input parameters. Deploy and test the different control flow paths in Remix.
- **Level 2:** Design and implement a Solidity smart contract with multiple functions, including internal and private functions. Demonstrate how these functions can be called and how visibility modifiers affect their accessibility.

Experiment 5: Contract Deployment

- **Level 1:** Deploy a pre-written simple smart contract (provided by the instructor) using MetaMask connected to the Ganache network. Observe the deployment transaction details (gas used, transaction hash, contract address).
- **Level 2:** Explore different deployment parameters in Remix IDE (e.g., setting gas limit and gas price). Deploy the same contract multiple times with varying gas settings and analyze the impact on deployment cost and confirmation time in Ganache.

Experiment 6: MetaMask and Remix IDE Interaction

- **Level 1:** Deploy a simple counter smart contract using Remix IDE on the Ganache network via MetaMask. Use the Remix interface to call the contract's functions (e.g., increment, decrement, get count) and observe the state changes reflected in both Remix and MetaMask (balance changes for transactions).
- **Level 2:** Deploy a more complex smart contract (e.g., a simple voting contract) using Remix and MetaMask. Interact with the contract through MetaMask's custom interaction interface (sending transactions to specific functions with appropriate arguments).

Experiment 7: Use of Geth - Installation and Account Management

- **Level 1:** Install the Geth Ethereum client on your local machine. Use Geth commands to create new Ethereum accounts and list the available accounts. Observe the keystore directory where private keys are stored.
- **Level 2:** Use Geth commands to export and import Ethereum account private keys. Understand the security implications of managing private keys. Connect the Geth console to a running private network (e.g., Ganache or a custom Geth network).

Experiment 8: Genesis Block Creation in Geth

- **Level 1:** Understand the structure of a Genesis Block JSON file. Modify a sample Genesis Block configuration (e.g., changing the initial coin distribution). Initialize a new Geth data directory using this modified Genesis Block.
- **Level 2:** Create a custom Genesis Block for a private Ethereum network with specific pre-allocated accounts, custom gas limit, and difficulty. Start a Geth node using this custom Genesis Block and connect to it using the Geth console.

Experiment 9: Interacting with a Private Geth Network

- **Level 1:** Start a Geth node using a previously initialized data directory. Use the Geth console to check the node's peer count and block number. Create a transaction to send Ether between two accounts within your private network using Geth commands.
- **Level 2:** Deploy a simple smart contract to your private Geth network using the Geth console and web3.js (or similar library). Interact with the deployed contract's functions using the Geth console.

Experiment 10: Exploring Ethereum Transaction Structure

- **Level 1:** Send a transaction (Ether transfer) using MetaMask on the Ropsten test network. Examine the transaction details on a block explorer (e.g., Etherscan for Ropsten) and identify key fields like `to`, `from`, `value`, `gas limit`, `gas price`, and `nonce`.
- **Level 2:** Construct and sign a raw Ethereum transaction using web3.js (or similar library) without relying on MetaMask. Broadcast this signed transaction to a test network (e.g., Ropsten) and analyze its details on a block explorer.

Experiment 11: Working with Smart Contract Events

- **Level 1:** Write a Solidity smart contract that emits events when certain actions occur (e.g., a value is updated). Deploy the contract in Remix and trigger the actions. Observe the emitted events in the Remix console.
- **Level 2:** Modify the previous contract to include indexed event parameters. Write a simple web3.js script (or use the Remix event listener) to filter and listen for specific events based on the indexed parameters.

Experiment 12: Understanding Smart Contract Security - Common Vulnerabilities (Part 1)

- **Level 1:** Study a simple smart contract with a known vulnerability (e.g., integer overflow/underflow - using an older Solidity version). Deploy the contract in Remix and attempt to exploit the vulnerability through function calls.
- **Level 2:** Research and demonstrate another common smart contract vulnerability (e.g., reentrancy - using a simplified example). Write a vulnerable contract and a separate "attacker" contract to exploit it on a local test network.

Experiment 13: Understanding Smart Contract Security - Common Vulnerabilities (Part 2)

- **Level 1:** Analyze a smart contract with access control implemented using `onlyOwner` modifier. Deploy the contract and attempt to call restricted functions from a non-owner account.
- **Level 2:** Explore the concept of gas limits and denial-of-service (DoS) attacks in smart contracts. Write a contract that could be susceptible to a simple gas-based DoS attack and demonstrate how it can be exploited.

Experiment 14: Interacting with Standard ERC-20 Tokens

- **Level 1:** Deploy a standard ERC-20 token contract (using OpenZeppelin library in Remix or a pre-written contract) on a local test network. Interact with the token contract's functions (e.g., `totalSupply`, `balanceOf`, `transfer`) using Remix.
- **Level 2:** Write a simple Solidity smart contract that interacts with the deployed ERC-20 token contract. Implement a function in your contract that allows users to spend a certain amount of the deployed tokens (requiring approval).

	Experiment 15: Introduction to Truffle Framework • **Level 1:** Install Truffle and Node.js. Create a new Truffle project. Understand the basic directory structure of a Truffle project (contracts, migrations, test). Compile a simple Solidity contract using Truffle commands. • **Level 2:** Write a simple test case for your smart contract using Truffle's testing framework (Chai and Mocha). Run the tests to ensure the contract functions as expected. Deploy your compiled contract to a local Ganache network using Truffle migrations.

Catalogue prepared by		
Recommended by the Board of Studies on		
Date of Approval by the Academic Council		

Course Code: CSE3433	**Course Title: Embedded & Decentralized Finance** **Type of Course: Program Core Theory & Lab Integrated**	**L-T-P-C**	2	0	2	3

Version No.		1.0
Course Pre-requisites	•	CSE3430 – Introduction to Fintech
Anti-requisites		NIL
Course Description		This course explores the integration of financial services into digital platforms through Embedded Finance and DeFi. It covers APIs, open banking, blockchain fundamentals, smart contracts, and cryptocurrencies.

		Students learn about decentralized platforms like DEXs, DAOs, and lending protocols. It highlights legal, security, and regulatory aspects, along with real-world case studies. The course also examines the convergence of DeFi with traditional finance and Web3 applications.
Course Object		The objective of the course is to familiarize the learners with the concepts of Embedded and Decentralized Finance and attain **Skill Development** through **Experiential Learning** techniques.
Course Out Comes		On successful completion of the course the students shall be able to: **CO1:** Understand the principles and applications of Embedded Finance and Decentralized Finance. (Understand) **CO2:** Explore blockchain technologies and smart contracts enabling DeFi. (Apply) **CO3**: Analyze the architecture and key components of Decentralized Finance (Apply) **CO4:** Evaluate security risks, vulnerabilities, and legal considerations in DeFi system. (Apply) **CO5:** Explain APIs, embedded wallets, and DeFi tools to develop basic solutions (Understand)
Course Content:		

Module 1	Introduction to Embedded Finance	Assignment			**5L+6P Sessions**
	Topics: Concept, Scope, and Evolution, APIs, Open Banking, and BaaS, Embedded Lending, Payments, Insurance, Case Studies: BNPL, Embedded Investment Platforms				
Module 2	Blockchain & Cryptocurrency Foundations	Assignment			**7L+6P Sessions**
	Topics: Blockchain Basics: Hashing, Consensus Mechanisms, Smart Contracts (Ethereum focus), Cryptocurrencies, Wallets, and Gas Fees, Public vs Private Blockchains				
Module 3	Decentralized Finance	Assignment			**6L+6PSessions**
	Topics: Concept, Benefits and Risks Associated with DeFi, Centralized vs Decentralized finance, DeFi Projects, DeFi future trends.				
Module 4	Risk Management and Regulation	Assignment			**6L+6P Sessions**
	Topics: Security Threats in DeFi: Smart Contract Bugs, Flash Loans, Legal and Compliance Issues: AML, KYC, Regulatory Perspectives (India, US, EU), Notable DeFi Hacks & Case Studies .				
Module 5	Future of Embedded and Decentralized Finance				**6L+6P**

	DeFi 2.0, Interoperability, Layer-2 Scaling, Real-World Asset Tokenization, Central Bank Digital Currencies (CBDCs), Web3 Integration and Embedded Wallets
	Project work/Assignment:
1.	2. **Assignment 1 on (Module 1 and Module 2)** 3. **Assignment 2 on (Module 3,4 and Module 5)**
	Text Book 1) Scarlett Sieber and Sophie Guibaud, "The Embedded Finance Handbook" , Wiley, 2023. 2) Campbell R. Harvey, "DeFi and the Future of Finance", Wiley, 2021. 3) Imran Bashir , "Mastering Blockchain", Packt, 2022.
	References **1. Antony Lewis, "The Basics of Bitcoins and Blockchains", Mango Publishing, 2021** **2. Turban Rainer Potter, Information Technology, John Wiely& Sons Inc,2012.** **3..** **Web Resources** W1. https://medium.com/search?q=decentralized+exchange W2. https://thefinancialbrand.com/111080/evolution-future-digital-banking-baastransformation/
	Lab Experiments **Experiment 1: Exploring Blockchain Basics** **Objective: Simulate a blockchain transaction and analyze block structure.** **Experiment 2: Create a Smart Contract for Token Transfer (ERC-20)** **Objective: Deploy a basic ERC-20 token on a local blockchain using Remix.** **Experiment 3: Build an Embedded Payment Flow using Stripe API** **Objective: Integrate an embedded payment gateway using Stripe's API.** **Experiment 4: Create and Test a Decentralized Lending Contract** **Objective: Build a simple DeFi lending smart contract with collateral logic.** **Experiment 5: Use a Decentralized Exchange (DEX)** **Objective: Swap tokens using Uniswap on testnet or via a demo platform.** **Experiment 6: Wallet Integration and Transaction Monitoring** **Objective: Build a Web3 app that connects MetaMask and shows wallet balance.** **Experiment 7: Implement a DAO Voting Mechanism** **Objective: Build and deploy a basic voting contract simulating DAO governance.** **Experiment 8: Explore Decentralized Insurance Use Case** **Objective: Create a smart contract for crop/weather-based insurance.** **Experiment 10: Analyze a Real DeFi Protocol**

	Objective: Study the architecture and working of Aave/Compound/Sushiswap.
	Topics relevant to development of "Employability": Real-world usage of APIs in FinTech, Building financial products with embedded APIs **Topics relevant to "PROFESSIONAL ETHICS":** Ethical handling of user data in embedded finance platform

Catalogue prepared by		
Recommended by the Board of Studies on		
Date of Approval by the Academic Council		

urse Code: **E3434**	**urse Title: Financial and Capital Markets** **pe of Course: Program Core** **eory & Lab Integrated**	**T-P-C**	2	0	2	3

rsion No.		
urse Pre-requisites	•	CSE3430 – Introduction to Fintech
ti-requisites		L
urse Description		This course provides a comprehensive overview of global financial markets, asset classes, and investment instruments. It covers the structure and functioning of various markets including money, equity, debt, derivatives, forex, and commodities. Students will learn about the roles of different participants, trading mechanisms, corporate actions, and market instruments such as bonds, mutual funds, and structured products. The course also delves into equity capital raising, trade life cycles, and global fund structures including open-ended and closed-ended investment vehicles. Emphasis is placed on both theoretical concepts and practical knowledge of global financial systems.
urse Object		The objective of the course is to familiarize the learners with the concepts of Embedded and Decentralized Finance and attain **Skill Development** through **Experiential Learning** techniques.
urse Out Comes		On successful completion of the course the students shall be able to: **CO1:** Understand the evolution of banking technology and its impact on modern banking operations.(Understand) **CO2:** Explain the role of delivery channels like ATMs, mobile banking, and UPI in digital banking systems.(Apply) **CO3: Evaluate** the effectiveness of risk, treasury, and data center management in

		banking operations.(Apply) **CO4: Analyze** the functioning of centralized banking and payment systems like RTGS, NEFT, and SFMS (Apply) **CO5:** Understand blockchain, cryptocurrency concepts, and analyze recent core banking software. (Understand)
urse Content:		

dule 1	Overview of Financial Markets And Assets Classes	signment			**10 Sessions**
	Topics: Cash and Money Markets, Bond markets, Foreign Exchange Markets, Equities Markets, Indices and Stocks, Derivatives Markets, Products and Settlement, Commodities Markets and Products, Saving and Investment Products, Mutual Fund and other Investment Products .				
dule 2	bal Equities Markets and Instruments	signment			**Sessions**
	Topics: Introduction to Equity Market-Introduction to Capital Markets, Equity Capital Markets, Raising Equity Through IPO, Raising Equity Through Private Sources, Equity buybacks, de-listing and reversion to a 'private' company. Equity Instruments & their characteristics-Stock Prices and Corporate Actions, Preference Shares, Depository Receipts, Rights Issues & Warrants, Convertibles, Equity Structured Products. Participants in the Equity Markets-Introduction and Role of the Buy Side, Buy Side Participants, Introduction and Role of Sell Side. Services and Participants in the Sell Side, Market Makers. Types of Equity Markets-Exchanges and Indices in the Equity Markets, Indices and their roles, Understand the difference between exchange and OTC markets, Types of weighted index, other indices and global indices, Electronic and Hybrid Markets and Order and Quote Driven Markets, Global Equity Markets. Trading of Equity Instruments-Equity Investments and its benefits and risks, Stock Quotations, Delivery or cash trading, Long and short positions, Leverage and Margin, Investing, trading and hedging, Placing Orders-limit orders, stop loss orders and GTD/GTC orders, Online and Offline Trading , Introduction to Trade Life Cycle, Clearing and Settlement				
dule 3	bal Foreign Exchange Markets and Instruments	signment			**Sessions**
	Topics: Introduction to Forex Market-What is foreign exchange market, Functions and purposes of the FX market, Introduction to types of Foreign Exchange Market. Participants in the foreign exchange market-Consumers & Travelers, Businesses, Investors & speculators, Commercial & Investment Banks, Government & Central Banks. Theories governing foreign exchange-Interest rate parity, Purchasing power parity, Nominal v/s real exchange rates, etc. Spot Market-Market organization, Quotation conventions, Direct and indirect prices, Cross rates, Value of a pip, Interpreting news and economic statistics, Delivery and operations. Forward Forex Market-Outright forward and swap deals, Relation between spot & forward markets, Quoting forward rates, Quoting swap points, Forward discounts and premiums, Forward forward transactions				
dule 4	bal Fixed Income (Bond) Markets and Instruments	signment			**essions**
	pics: erview of Debt Capital Markets-Characteristics of Debt Capital Markets, The differences between equity and debt products, The differences between loans and bonds, Hybrid securities, Securitization. Bond-An Introduction- Bond definition, Bond Issuer & Bond Investor, Types of bond, Bond				

	characteristics, Zero Coupon Bond, Price/yield relationship, Government bond markets, The Eurobond market .				
odule 5	obal Funds				essions
	oduction -Potential advantages and disadvantages of collective investment, Difference between active and passive management .Open-Ended/Mutual Funds-Characteristics and different types of open-ended fund / mutual fund: • US • Europe , Purpose and principal features of the Undertakings for Collective Investment in Transferable Securities (UCITS) directive in European markets .Closed Ended Investment Companies-Characteristics of closed-ended investment companies, share classes, Meaning of the discounts and premiums in relation to the pricing of closed-ended investment companies , How closed-ended investment companies' shares are traded. Off shore and On-shore Global financial centers				
	oject work/Assignment:				
12	**13. Assignment 1 on (Module 1 and Module 2)** 14. **Assignment 2 on (Module 3,4 and Module 5)**				
	Text Book 12) Financial Markets and Institutions 7th Edition By Anthony Saunders and Marcia Cornett, Ninth Edition, McGraw Hill Education, 2024. 13) Mishkin, F. S., & Eakins, S. G. (2018) "*Financial Markets and Institutions"* (9th Edition). Pearson Education.				
	References **1. Gordon, E. & Natarajan, K. (2022)** ***Financial Markets and Services*** **(Latest Edition). Himalaya Publishing House.** **2.**Bhole,L.M.&Mahakud,J.(2017) Financial Institutions and Markets: Structure, Growth, and Innovations (5th Edition). McGraw Hill Education**.** **Web Resources** W10. https://www.ibm.com/industries/banking-financial-markets/resources/omnichannelbanking-paper/ W11. https://thefinancialbrand.com/111080/evolution-future-digital-banking-baastransformation/				
	Design and Develop the following Banking Software using the appropriate technologies: **■ Mobile Banking** **▪ Balance Enquiry ▪ Cheque book Request ▪ Stop Cheque ▪ Credit/Debit Notification ▪ Bill Payment** **■ Internet Banking** **▪ Electronic Funds Transfer ▪ Account Management ▪ Loan Application ▪ Registering of new bank services ▪ Customer Information Management** **■ATM system** **▪ Balance Enquiry ▪ Withdrawal ▪ Deposit ▪ Pin change ▪ Mini statement**				
	Topics relevant to development of "Employability": Real time Data Analysis for Banking Technology. **Topics relevant to "PROFESSIONAL ETHICS":** Mobile, Internet Banking for Project Development.				
Catalogue prepared by					

Recommended by the Board of Studies on		
Date of Approval by the Academic Council		

urse Code: E3435	**urse Title: Blockchain Development and Programming** **pe of Course: Program Core** **eory & Lab Integrated**	**Γ-P-C**	2	0	2	3

rsion No.		
urse Pre-requisites	•	CSE3432 – Blockchain Technology
ti-requisites		L
urse Description		This course provides a comprehensive introduction to the fundamental concepts and applications of blockchain technology. Students will explore the principles of decentralization, understand the mechanics of Bitcoin, delve into the world of smart contracts and alternative cryptocurrencies, and gain practical experience in developing and deploying smart contracts using industry-standard tools. The course emphasizes both the theoretical underpinnings and the practical implementation of blockchain solutions.
urse Object		The objective of the course is to familiarize the learners with the concepts of Blockchain Development and Programming and attain **Skill Development** through **Experiential Learning** techniques.
urse Out Comes		On successful completion of the course the students shall be able to: **CO1:** Understand blockchain and decentralization principles. (Understand) **CO2:** Analyze Bitcoin's operation and transactions. (Apply) **CO3**: Apply smart contract concepts and explore altcoins. (Apply) **CO4:** Utilize Truffle for smart contract development and deployment. (Apply)
urse Content:		

ɔdule 1	**Fundamentals of Blockchain and Decentralization**	signment			**6L+6P Sessions**
	Introduction to Blockchain: the growth of blockchain technology, the history of blockchain and Bitcoin, distributed systems, blockchain, consensus. Decentralization: decentralization using blockchain, methods of decentralization, routes to decentralization, blockchain and full ecosystem decentralization, pertinent terminology, platforms for decentralization, innovative trends.				

odule 2	coin: The First Cryptocurrency	signment				+8P Sessions
	Topics: Introduction to Bitcoin: bitcoin — an overview, cryptographic keys, transactions, blockchain, mining. The Bitcoin Network and Payments: the bitcoin network, wallets, bitcoin payments, innovation in bitcoin, advanced protocols, bitcoin investment and buying and selling Bitcoin. Bitcoin Clients and APIs: bitcoin client installation, experimenting further with bitcoin-cli, bitcoin programming					
odule 3	yond Bitcoin: Altcoins and Smart Contracts	signment				L+8PSessions
	Topics: Consensus Algorithms: introducing the consensus problem, analysis and design, classification, algorithms, choosing an algorithm. Alternative Coins: introducing altcoins, theoretical foundations, difficulty adjustment and retargeting algorithms, bitcoin limitations, extended protocols on top of bitcoin, development of altcoins, Initial Coin Offerings (ICOs). Smart Contracts: history, definition, ricardian contracts, smart contract templates, oracles, deploying smart contracts, the DAO. .					
odule 4	**Ethereum and the Decentralized Web (Web3)**	signment				+8P Sessions
	pics: Ethereum 101: ethereum – an overview, the ethereum network, components of the ethereum ecosystem, Ethereum Virtual Machine (EVM), smart contracts, ethereum development environment. Further Ethereum: blocks and blockchain, wallets and client software, nodes and miners, APIs, tools, and DApps, supporting protocols, programming languages. Introducing Web3: contract deployment, exploring Web3 with Geth					
	oject work/Assignment:					
15	**16. Assignment 1 on (Module 1 and Module 2)** 17. **Assignment 2 on (Module 3 and Module 4)**					
	Text Books 1. Banafa, A. (2024). *Blockchain technology and applications*. River Publishers. 2. Ramachandran, M. (2025). *Blockchain engineering: Secure, sustainable frameworks for healthcare applications*. Springer. 3. Tanwar, S. (2022). *Blockchain technology: From theory to practice*. Springer. 4. Vyas, S., Shukla, V. K., Gupta, S., & Prasad, A. (Eds.). (2022). *Blockchain technology: Exploring opportunities, challenges, and applications*. CRC Press.					
	References 1. Chuen, D. L. K. (Ed.). (2024). *Handbook of digital currency: Bitcoin, innovation, financial instruments and big data* (2nd ed.). Academic Press. 2. Idrees, S. M. & Nowostawski, M. (Eds.). (2023). *Transformations through blockchain technology: The new digital revolution*. Springer.					

3. Jena, A. K., Panda, S. K., & Swain, S. K. (Eds.). (2022). *Blockchain technology: Applications and challenges* (Vol. 203). Springer.
4. Maleh, Y., Zhang, J., & Hansali, A. (2024). *Advances in emerging financial technology and digital money*. Routledge.
5. Rahman, H. (Ed.). (2025). *Blockchain technology applications in knowledge management*. IGI Global

Web Resources

W1. Blockgeeks. Retrieved from https://www.google.com/search?q=blockgeeks.com
W2. Bitcoin.org. Retrieved from https://bitcoin.org/
W3. CoinDesk. Retrieved from https://www.coindesk.com/
W4. Ethereum.org. Retrieved from https://ethereum.org/
W5. Investopedia. Retrieved from https://www.investopedia.com/
W6. Medium. Retrieved from https://medium.com/
W7. Solidity Documentation. Retrieved from https://docs.soliditylang.org/
W8. Truffle Suite Documentation. Retrieved from https://trufflesuite.com/docs
W9. Web3.js Documentation. Retrieved from https://web3js.readthedocs.io/
W10. GitHub. Retrieved from https://github.com/

Experiment 1: Integrated Development Environments (IDEs) for Smart Contracts

- **Level 1:** Explore the features and interface of Remix IDE. Deploy a simple "Hello World" smart contract on the in-browser JavaScript VM. Observe the transaction details and contract interaction options.
- **Level 2:** Install and configure MetaMask browser extension. Connect MetaMask to the Remix IDE. Deploy the same "Hello World" contract to the Ganache private network via MetaMask. Examine the transaction process in both Remix and MetaMask.

Experiment 2: MetaMask in a Private Network

- **Level 1:** Set up a local Ganache private network. Add a custom network in MetaMask, configuring the RPC URL and Chain ID to connect to your Ganache instance. Create a new account in MetaMask and observe its balance.
- **Level 2:** Deploy a simple token contract (e.g., ERC-20 minimal) using Remix IDE and MetaMask on your private Ganache network. Transfer some tokens between the accounts you created in MetaMask and observe the balance changes.

Experiment 3: Smart Contract with Solidity - Basic Data Types and Structures

- **Level 1:** Write a Solidity smart contract that declares and initializes variables of different basic data types (uint, string, bool, address). Implement functions to read and modify these variables. Deploy and interact with the contract in Remix IDE.
- **Level 2:** Create a Solidity smart contract that utilizes structs and arrays. Implement functions to add, retrieve, and update elements within these data structures. Deploy and test the contract with various inputs in Remix IDE.

Experiment 4: Smart Contract with Solidity - Control Flow and Functions

- **Level 1:** Write a Solidity smart contract that uses `if-else` statements and `for` loops within its functions. Implement a function that performs a simple calculation based on input parameters. Deploy and test the different control flow paths in Remix.
- **Level 2:** Design and implement a Solidity smart contract with multiple functions, including internal and private functions. Demonstrate how these functions can be called

and how visibility modifiers affect their accessibility.

Experiment 5: Contract Deployment

- **Level 1:** Deploy a pre-written simple smart contract (provided by the instructor) using MetaMask connected to the Ganache network. Observe the deployment transaction details (gas used, transaction hash, contract address).
- **Level 2:** Explore different deployment parameters in Remix IDE (e.g., setting gas limit and gas price). Deploy the same contract multiple times with varying gas settings and analyze the impact on deployment cost and confirmation time in Ganache.

Experiment 6: MetaMask and Remix IDE Interaction

- **Level 1:** Deploy a simple counter smart contract using Remix IDE on the Ganache network via MetaMask. Use the Remix interface to call the contract's functions (e.g., increment, decrement, get count) and observe the state changes reflected in both Remix and MetaMask (balance changes for transactions).
- **Level 2:** Deploy a more complex smart contract (e.g., a simple voting contract) using Remix and MetaMask. Interact with the contract through MetaMask's custom interaction interface (sending transactions to specific functions with appropriate arguments).

Experiment 7: Use of Geth - Installation and Account Management

- **Level 1:** Install the Geth Ethereum client on your local machine. Use Geth commands to create new Ethereum accounts and list the available accounts. Observe the keystore directory where private keys are stored.
- **Level 2:** Use Geth commands to export and import Ethereum account private keys. Understand the security implications of managing private keys. Connect the Geth console to a running private network (e.g., Ganache or a custom Geth network).

Experiment 8: Genesis Block Creation in Geth

- **Level 1:** Understand the structure of a Genesis Block JSON file. Modify a sample Genesis Block configuration (e.g., changing the initial coin distribution). Initialize a new Geth data directory using this modified Genesis Block.
- **Level 2:** Create a custom Genesis Block for a private Ethereum network with specific pre-allocated accounts, custom gas limit, and difficulty. Start a Geth node using this custom Genesis Block and connect to it using the Geth console.

Experiment 9: Interacting with a Private Geth Network

- **Level 1:** Start a Geth node using a previously initialized data directory. Use the Geth console to check the node's peer count and block number. Create a transaction to send Ether between two accounts within your private network using Geth commands.
- **Level 2:** Deploy a simple smart contract to your private Geth network using the Geth console and web3.js (or similar library). Interact with the deployed contract's functions using the Geth console.

Experiment 10: Exploring Ethereum Transaction Structure

- **Level 1:** Send a transaction (Ether transfer) using MetaMask on the Ropsten test network. Examine the transaction details on a block explorer (e.g., Etherscan for Ropsten) and identify key fields like `to`, `from`, `value`, `gas limit`, `gas price`, and `nonce`.
- **Level 2:** Construct and sign a raw Ethereum transaction using web3.js (or similar library) without relying on MetaMask. Broadcast this signed transaction to a test network (e.g.,

Ropsten) and analyze its details on a block explorer.

Experiment 11: Working with Smart Contract Events

- **Level 1:** Write a Solidity smart contract that emits events when certain actions occur (e.g., a value is updated). Deploy the contract in Remix and trigger the actions. Observe the emitted events in the Remix console.
- **Level 2:** Modify the previous contract to include indexed event parameters. Write a simple web3.js script (or use the Remix event listener) to filter and listen for specific events based on the indexed parameters.

Experiment 12: Understanding Smart Contract Security - Common Vulnerabilities (Part 1)

- **Level 1:** Study a simple smart contract with a known vulnerability (e.g., integer overflow/underflow - using an older Solidity version). Deploy the contract in Remix and attempt to exploit the vulnerability through function calls.
- **Level 2:** Research and demonstrate another common smart contract vulnerability (e.g., reentrancy - using a simplified example). Write a vulnerable contract and a separate "attacker" contract to exploit it on a local test network.

Experiment 13: Understanding Smart Contract Security - Common Vulnerabilities (Part 2)

- **Level 1:** Analyze a smart contract with access control implemented using `onlyOwner` modifier. Deploy the contract and attempt to call restricted functions from a non-owner account.
- **Level 2:** Explore the concept of gas limits and denial-of-service (DoS) attacks in smart contracts. Write a contract that could be susceptible to a simple gas-based DoS attack and demonstrate how it can be exploited.

Experiment 14: Interacting with Standard ERC-20 Tokens

- **Level 1:** Deploy a standard ERC-20 token contract (using OpenZeppelin library in Remix or a pre-written contract) on a local test network. Interact with the token contract's functions (e.g., `totalSupply`, `balanceOf`, `transfer`) using Remix.
- **Level 2:** Write a simple Solidity smart contract that interacts with the deployed ERC-20 token contract. Implement a function in your contract that allows users to spend a certain amount of the deployed tokens (requiring approval).

Experiment 15: Introduction to Truffle Framework

- **Level 1:** Install Truffle and Node.js. Create a new Truffle project. Understand the basic directory structure of a Truffle project (contracts, migrations, test). Compile a simple Solidity contract using Truffle commands.
- **Level 2:** Write a simple test case for your smart contract using Truffle's testing framework (Chai and Mocha). Run the tests to ensure the contract functions as expected. Deploy your compiled contract to a local Ganache network using Truffle migrations.

Topics relevant to development of "Employability": Hands-on experience with blockchain platforms, smart contract development, and decentralized applications

Topics relevant to "PROFESSIONAL ETHICS": Understanding ethical considerations in blockchain use, such as data privacy, transparency, and responsible innovation, promotes integrity in decentralized systems

Catalogue prepared by		
Recommended by the Board of Studies on		
Date of Approval by the Academic Council		

urse Code: E3436	**urse Title: Statistics and Data Analysis for Finance** **pe of Course: Program Core** **eory & Lab Integrated**	**T-P-C**	2	0	2	3

rsion No.		
urse Pre-requisites	•	CSE3430 – Introduction to Fintech
ti-requisites		L
urse Description		This course provides a comprehensive understanding of the theory and practice of data analysis in the Accounting and wider business domains. Students who complete this course should develop the skills to apply and interpret data-based initiatives that address real-world problems across many financial activities such as financial accounting, management accounting, taxation, auditing and corporate finance.
urse Object		The objective of the course is to familiarize the learners with the concepts of Statistics and Data Analysis for Finance and attain **Skill Development** through **Experiential Learning** techniques.
urse Out Comes		On successful completion of the course the students shall be able to: **CO1:** Understand the strategic processes, benefits and challenges. (Understand) **CO2:** Apply preprocessing techniques to business datasets (Apply) **CO3**: Apply statistical techniques to the datasets . (Apply) **CO4:** Analyse data protection, data privacy and other ethical issues. (Apply)
urse Content:		

dule 1	**Strategic Data Management in in Finance**	signment				**6L+6P Sessions**
	Topics: Challenges in Human Decision Making, Data Analytics Processes - Introduction to Process Models, Financial Big Data for Competitive Advantage, Strategic Data Management in Finance, Management Challenges in Data Driven Environments, Developing a Data Driven Culture					
dule 2	**ploratory Data Analysis**	signment				**+8P Sessions**

	Topics: Exploratory Data Analysis, Data Preparation – Normalization, Binning, Sampling Descriptive Statistics, Data Quality Issues - Missing Values, Outliers, Visualizing Relationships Between Features, Measuring Covariance and Correlation, Data Distributions and Confidence Intervals, • Simple Linear Regression, Correlation Coefficient, Calculation of Regression parameters				
odule 3	**atistics for Big data**	signment			**L+8PSessions**
	Topics: Effect size, Statistical power and sample size, Effect of Variation, Hypothesis testing Interpret outputs from statistical software to analyse patterns in accounting data for signalling unexpected fluctuations e.g. Fraud Analysis, detecting anomaly transactions etc.. , Time Series basics, Decomposition of Time Series, Seasonality, Linear Trend models, Smoothing models, Interpret outputs from statistical software to support prediction of accounting data.				
odule 4	**Dimension Reduction & Data Ethics**	signment			**+8P Sessions**
	pics: Factor Analysis, Principal Component Analysis (PCA), Interpret outputs from statistical software to analyse reasons behind fluctuations in accounting data e.g. defining cost drivers in Activity Based Costing etc, Data Ethics in Finance, Data Legislation GDPR, Data and Statistical reporting				
	oject work/Assignment:				
18	**19. Assignment 1 on (Module 1 and Module 2)** 20. **Assignment 2 on (Module 3 and Module 4)**				
	Text Books 5. Foster Provost, Tom Fawcett, Data Science for Business, O'Reilly Media, 2023. 6. Wes McKinney, Python for Data Analysis, O'Reilly Media (2nd Ed.), 2022. 7. Peter J. Brockwell, Richard A., Introduction to Time series and Forecasting, Springer, 2023. 8. W. Gregory Voss, Hélène J. Lefebvre, Data Ethics in the Digital Age, Springer, 2023.				
	References 6. Mark J. Bennett, Dirk L. Hugen, Financial Analytics with R: Building a Laptop Laboratory for Data Science, Cambridge University Press, 2021. 7. Thomas H. Davenport, Analytics at Work: Smarter Decisions, Better Results, Harvard Business Press, 2022. 8. Peter Bruce, Andrew Bruce, Peter Gedeck, Practical Statistics for Data Scientists, O'Reilly Media (2nd Ed., 2023. 9. Satish Kumar, Principles and Practice of Multi-dimensional Data Analysis, Wiley, 2022. **Web Resources** W1. https://link.springer.com/book/10.1007%2F978-3-030-01279-3 W2. https://link.springer.com/book/10.1007%2F978-1-4939-2122-5 W3. https://link.springer.com/book/10.1007%2F978-3-319-55444-0				
	Strategic Data Management in Finance • Identify and discuss the challenges in Human Decision Making, particularly relating to large datasets.				

	• Discuss the benefits and challenges of utilising Process Models to manage Finance Data Analysis projects • Identify and discuss the strategic benefits to be derived from Financial Big Data . Identify and discuss the management challenges in leveraging the benefits of Big Data for strategic competitive advantage Exploratory Data Analysis • Discuss the process of data cleaning and preparation – e.g. Normalization, Binning, Sampling • Apply and Evaluate key descriptive statistics, including Covariance and Correlation, in a data set for large business datasets • Discuss solutions to overcome data quality issues in Data Analysis projects - missing values, outliers etc. Apply and Evaluate methods for visualizing relationships between features Statistics for Big data • Explain relationships between sample size, effect size, statistical power • Describe and Evaluate measures of variation for large datasets • Describe hypothesis testing and evaluate outputs from hypothesis tests performed using software such as Excel, R and Python etc. Interpret outputs from statistical software to analyse patterns in accounting data for signaling unexpected fluctuations - fraud analysis, detection of anomaly transactions etc. Dimension reduction • Explain PCA and factor analysis and discuss its uses in the analysis of large financial datasets Interpret outputs from statistical software to analyse reasons behind fluctuations in accounting data e.g. defining cost drivers in Activity Based Costing etc. Data Ethics & Legal Considerations • Identify and discuss the ethical issues surrounding the use of data analytics in finance • Demonstrate an understanding of Data Legislation GDPR and its impact on data analytics Discuss the societal impacts of the increasing use of Data Analysis techniques in Finance and Business
	Topics relevant to development of "Employability": Hands-on experience with PCA, Hypothesis test. **Topics relevant to "PROFESSIONAL ETHICS":** Understanding ethical considerations and legal considerations

Catalogue prepared by		
Recommended by the Board of Studies on		
Date of Approval by the Academic Council		

Course Code: CSE3437	**Course Title:** Financial Regulations and Compliances **Type of Course:** Theory	**L- T- P- C**	3	**0**	0	**3**

Version No.	1.0			
Course Pre-requisites				
Anti-requisites	**NIL**			
Course Description	This course provides a comprehensive understanding of the evolving landscape of financial technology (FinTech) regulations and the growing importance of Regulatory Technology (RegTech). It examines the historical development of FinTech regulation, analyzes the specific regulatory framework governing FinTech innovations in India, and explores the application of regulatory sandboxes. Furthermore, the course evaluates the challenges associated with the adoption and implementation of RegTech solutions in the financial services industry. By the end of this course, students will gain a strong foundation in the key regulatory considerations shaping the FinTech ecosystem and the role of technology in navigating this complex environment.			
Course Objective	The objective of this course is to equip learners with a practical understanding of FinTech regulations and RegTech, and to develop their analytical and problem-solving skills through active participation in learning activities, thereby enhancing their employability in the evolving financial technology sector.			
Course Outcomes	On successful completion of the course the students shall be able to: CO1: **Understand** the evolution of FinTech regulation and the emergence of RegTech. CO2: **Explain** the regulations governing FinTech within the Indian context. CO3: **Describe** the purpose and function of regulatory sandboxes. CO4: **Analyze** the challenges in adopting RegTech solutions.			
Course Content:				
Module 1	**Introduction to FinTech regulation and RegTech**	Assignment	RegTech Ecosystem	**10 Sessions**
FinTech Regulation, Evolution of RegTech- RegTech 1.0, RegTech 2.0, RegTech 3.0, RegTech ecosystem- Financial institutions, start-ups, and regulators. The future of Regtech and the technologies impacting it				
Module 2	Regulations governing FinTech in India	Assignment	Regulations	**10 Sessions**
Regulation of mobile money, Regulation of smart contracts, Regulation of Robo-Advisory services, legal and regulatory implications of cryptocurrencies, Payment and Settlements System Act 2007, Master direction on Issuance and operation of prepaid payments instruments. NPCI guidelines governing UPI payments, Master direction-NBFC, Guidelines regulating P2P lending platforms, payment aggregators/intermediaries, payment banks, Anti money-laundering regulations, Data privacy and protection.				
Module 3	Regulatory Sandboxes	Assignment	Regulatory Sandboxes	**9 Sessions**
Introduction, what is regulatory sandbox-Covered FinTech products and eligible participants, parameters, regulatory safe harbour, Post sandbox engagement. Benefits-Participant-regulator dialogue, reduced time and cost of market penetration, stronger appeal to stakeholders, market signalling. Shortcoming-Multi-tiered regimes, Pre-judging innovative value, scalability, race to the bottom. Regulatory sandbox in India, China, USA, Europe and other countries.				

Module 4	Challenges, Future and Use Case	Assignment	Complaince	**9 Sessions**

Risks and challenges of RegTech adoption-Procurement and approval process, Preference for large and established players, Fragmented markets, Regulatory uncertainty, Concentration risk, Data protection security and cyber threats. Future Trends in RegTech-Quantitative Regulation, Machine readable regulation, Agile Regulation, Regulatory Sandboxes, International regulation. Compliance, Identity management and control, risk management, Regulatory reporting, Transaction monitoring, Trading in markets

Targeted Application & Tools that can be used:

Text Book(s):

1. Madir, J. (Ed.). (2024). *FinTech: Law and Regulation* (3rd ed.). Edward Elgar Publishing.
2. Securities and Exchange Board of India. (2014). *Consultation paper on crowdfunding in India*. https://www.sebi.gov.in/sebi_data/attachdocs/1403005615257.pdf
3. Reserve Bank of India. (2025, April 9). *Enabling Framework for Regulatory Sandbox* https://www.rbi.org.in/Scripts/PublicationReportDetails.aspx?UrlPage=&ID=938

Reference(s):

1. Ren, D. (2018). Tightening regulations make FinTechs easy takeover targets for banks stepping up digitalisation drive. *SCMP*. Retrieved from https://www.scmp.com/business/companies/article/2159718/tightening-regulations-make-fintechs-easy-takeover-targets-banks
2. Zetzsche, D. A., Buckley, R. P., Arner, D. W., & Barberis, J. N. (2017). *From FinTech to TechFin: The regulatory challenges of data-driven finance* (University of Hong Kong Faculty of Law Research Paper No. 2017/007). http://dx.doi.org/10.2139/ssrn.2959925
3. Magnuson, W. J. (2017). *Regulating Fintech* (Texas A&M University School of Law Legal Studies Research Paper No. 17-55). Retrieved from https://ssrn.com/abstract=3027525
4. Sethi, V. (n.d.). *Fintech & Regtech - your definitive guide on the convergence of finance, technology and regulation* (p. 3) [Kindle Edition]. Max Krish Publishers.
5. Lui, A., & Ryder, N. (Eds.). (2023). *FinTech, Artificial Intelligence and the Law: Regulation and Crime Prevention*. Routledge.
6. Shrier, D. L. (Ed.). (2022). Global Fintech: Financial Innovation in the Connected World. MIT Press.
7. McGurk, B. KC., & Reichenbach, S. (2024). Financial Services Law and Distributed Ledger Technology: Regulating Cryptoassets and Decentralised Finance. Edward Elgar Publishing.
8. Buckley, R. P., Arner, D. W., & Zetzsche, D. A. (2023). Fintech finance technology and regulation. Cambridge University Press.
9. Justin, M. S. M., et al. (Eds.). (2024). Examining Global Regulations During the Rise of Fintech. IGI Global.
10. El Dimachki, M. (2024). Fintech Regulation In Practice. Kogan Page.
11. Madir, J. (Ed.). (2024). FinTech: Law and Regulation (3rd ed.). Edward Elgar Publishing.
12. McGurk, B. KC., & Reichenbach, S. (2024). Financial Services Law and Distributed Ledger Technology: Regulating Cryptoassets and Decentralised Finance. Edward Elgar Publishing.
13. Risk Books. (2025). Regtech, Suptech and Beyond: Innovation in Financial Services.
14. Gupta, S., et al. (Eds.). (2024). Integrating RegTech Solutions for Industry 4.0. IGI Global.

Websites:

1. **Financial Conduct Authority.** (n.d.). *FinTech*. Retrieved from https://www.fca.org.uk/firms/innovation/fintech
2. **Securities and Exchange Board of India.** (n.d.). *Homepage*. Retrieved from https://www.sebi.gov.in/

3. **Reserve Bank of India.** (n.d.). *Homepage*. Retrieved from https://www.rbi.org.in/
4. **Financial Industry Regulatory Authority.** (n.d.). *FinTech*. Retrieved from https://www.finra.org/rules-guidance/key-topics/fintech
5. **International Financial Services Centres Authority.** (n.d.). *FinTech Hub*. Retrieved from https://ifsca.gov.in/FinTechHub2023/ifsca.gov.in/Pages/Contents/FinnTechHub.html
6. **Federal Trade Commission.** (n.d.). *Fintech*. Retrieved from https://www.ftc.gov/business-guidance/credit-finance/fintech
7. **Deloitte Luxembourg.** (n.d.). *Regtech Universe*. Retrieved from https://www.deloitte.com/lu/en/Industries/technology/analysis/regtech-companies-compliance.html
8. **Apiax.** (n.d.). *The Ultimate RegTech Guide*. Retrieved from https://www.apiax.com/resources/guides/regtech-guide/
9. **Ascent RegTech.** (n.d.). *Homepage*. Retrieved from https://www.ascentregtech.com/
10. **CUBE Global.** (n.d.). *Homepage*. Retrieved from https://cube.global/
11. **ACA Group.** (n.d.). *ComplianceAlpha RegTech Solutions*. Retrieved from https://www.acaglobal.com/our-solutions/compliancealpha
12. **FinTech Futures.** (n.d.). *RegTech Archives*. Retrieved from https://www.fintechfutures.com/category/regtech/

Catalogue prepared by	
Recommended by the Board of Studies on	
Date of Approval by the Academic Council	

urse Code: CAI3427	urse Title: Language Models for Text Mining ype of Course: Discipline Elective - Theory & Integrated Laboratory	-P-C	2			2
rsion No.						
urse Pre-requisites	CSE1700 – Essentials of AI					
ti-requisites						
urse Description	This course introduces the basics of Text Mining and Natural Language Processing. The course will teach students different concepts such as text mining, NLP, Sequence Labeling, etc. Topics: Text Mining, NLP, Tokenization, Lemmatization, Stemming, One-hot encoding, Language modelling, Bag-of-words, Term-document Matrix, Cosine similarity, Viterbi Algorithm, etc.					
urse Objectives	The objective of the course is EMPLOYBILITY of student by using EXPERIENTIAL LEARNING techniques.					
urse OutComes	On successful completion of this course the students shall be able to: 1. **Process** text data to derive information from text. [Apply] 2. **Apply** insights from textual information to real-world business. [Apply] 3. **Develop** solutions for a particular NLP problem using different machine learning and deep learning techniques. [Apply] 4. **Utilize** different NLP tools and packages. [Apply]					

Course Content:

dule 1	xt Mining	versarial Quiz Tests	dule Tests	No. of Sessions: 09
Introduction to Text Mining. Text Mining vs. NLP. Text Mining Algorithms. Steps in Text Mining - Extraction, Preprocessing, Analysis and Evaluation. Lexical Resource Creation (NEW). Data collection. String Manipulation to Clean Data. Natural Language Processing. Research Paradigms in NLP. Sequential Data. Sequence Labeling (NEW). Viterbi Algorithm (NEW). Corpus. Building a HMM using a Corpus (NEW). Unknown word handling (NEW).				
dule 2	xt Preprocessing	versarial Quiz Tests	dule Tests	No. of sessions: 06
Introduction to Preprocessing. Tokenization. Stop Words Removal. Lemmatization and Stemming. PoS Tagging. Integer Encoding. Padding. One-Hot Encoding.				
dule 3	xt Representations	versarial Quiz Tests	dule Tests	No. of sessions: 08
Language Modeling. N-Gram Language Model. Bag-of-Words Model. Term-Document Matrix. Term Frequency. Inverse Document Frequency. TF-IDF. Cosine Similarity. Naive Bayes Classifier using Bag-of-Words. Topic Modeling. Latent Semantic Analysis. Singular Value Decomposition. Truncated SVD and Topic Vector. LDA Algorithm.				
dule 4	tural Language Processing with Keras	versarial Quiz Tests	dule Tests	No. of Sessions: 06
Word Embeddings vs. One-Hot Encoding. Contextual Bag of Words (CBOW). Skipgram. Deep Learning				

for Document Classification.

List of Laboratory Tasks:

Experiment No. 1: File Handling
Level 1: Read text files using Python and extract meaningful content.
Level 2: Parse text files using Python to preprocess the data for NLP tasks.

Experiment No. 2: Introduction to NLP Tools
Level 1: Install and use NLTK for basic text processing.
Level 2: Install and use SpaCy for tokenization, PoS tagging, and Named Entity Recognition.

Experiment No. 3: Corpus Cleaning Techniques
Level 1: Use NLTK for corpus cleaning techniques such as tokenization, stopword removal, and stemming.
Level 2: Prepare cleaned text data for downstream NLP tasks like classification or translation.

Experiment No. 4: Word Vector Usage
Level 1: Download and use pre-trained word vectors (e.g., Word2Vec, GloVe, or FastText).
Level 2: Compute similarity between two words, find the most similar word, and complete word analogies (e.g., king - man + woman = queen).

Experiment No. 5 & 6: Language Identification
Level 1: Build a simple language identifier using Bag-of-Words (BoW) features.
Level 2: Predict the language of a given text using the trained model.

Experiment No. 7 & 8: Lexical Simplification
Level 1: Implement a lexical simplifier to replace complex words with simpler alternatives.
Level 2: Generate a simplified version of a given word or sentence while preserving meaning.

Experiment No. 9 & 10: Sentiment Analysis
Level 1: Implement a basic sentiment classifier using a lexicon-based or machine learning approach.
Level 2: Compare the performance of an existing sentiment classifier (e.g., VADER, TextBlob, or a pre-trained Transformer model).

Experiment No. 11: Named Entity Recognition (NER)
Level 1: Extract named entities from a text using NLTK.
Level 2: Extract named entities using SpaCy and compare results.

Experiment No. 12 & 13: Implement a Hidden Markov Model (HMM)
Level 1: Implement a generic HMM for sequence prediction.
Level 2: Calculate the forward probability of a given sequence using HMM.

Experiment No. 14: Linguistic HMM
Level 1: Develop a Hidden Markov Model (HMM) for NLP tasks such as PoS tagging.
Level 2: Evaluate the performance of the HMM on a specific NLP task (e.g., Named Entity Recognition or Chunking).

Experiment No. 15: Machine Translation
Level 1: Implement Machine Translation (MT) using a pre-trained model from Hugging Face Transformers.
Level 2: Evaluate the quality of MT output via Round-Trip Translation (translate text to another language and back to check accuracy).

Targeted Application & Tools that can be used:

1. Google Colab
2. Python IDEs like PyCharm

oject work/Assignment: Mention the Type of Project /Assignment proposed for this course

1. Group project on some NLP Task like text classification (Creating a Simple Text Classifier: Use Scikit-learn to classify positive vs. negative reviews from a dataset), sentiment analysis, etc.

xtbook(s): 1. Daniel Jurafsky, James H. Martin. "Speech and Language Processing: An Introduction to Natural Language Processing", Computational Linguistics and Speech, Pearson Publication, 2025 (3rd Edition Draft). **2.** Aditya Joshi, Pushpak Bhattacharyya. "Natural Language Processing", Wiley Publication, 2023 (1st Edition).
ferences: R1. Chris Manning and Hinrich Schutze, "Foundations of Statistical Natural Language Processing", 1st Edition, MIT Press. 1999. R2. Pawan Goyal. "Natural Language Processing". 1st Edition, 2016. **eblinks** . E-Book link or R2: https://drive.google.com/file/d/10nbwAJd-dv6htOOZVBgAvLd1WscI0RqC/view . Web Resource for T1: https://web.stanford.edu/~jurafsky/slp3/ - VERY VERY IMPORTANT!!! **W3**. NPTEL Courses: https://nptel.ac.in/courses/106106211 CMI), https://nptel.ac.in/courses/106105158 (IIT Kgp), https://nptel.ac.in/courses/106101007 (IITB), https://nptel.ac.in/courses/106105572 (IIT Kgp - NEW)

urse Code: CAI3428	**urse Title: Practical Deep Learning with TensorFlow** **pe of Course: Discipline Elective - Theory & Integrated Laboratory**	**T-P- C**				
rsion No.						
urse Pre-requisites	CSE1700 – Essentials of AI					
ti-requisites	L					
urse Description	This course introduces students to the concepts of deep neural networks and state of the art approaches to develop deep learning models. In this course students will be given an exposure to the details of neural networks as well as deep learning architectures and to develop end-to-end models for such tasks. It will help to design and develop an application-specific deep learning models and also provide the practical knowledge handling and analyzing end user realistic applications.					
urse Objective	This course is designed to improve the learners EMPLOYABILITY SKILLS by using EXPERIENTIAL LEARNING techniques.					
urse Outcomes	On successful completion of this course the students shall be able to: 5. Implement backpropagation and gradient descent techniques to train neural networks effectively. (Apply) 6. Build and train deep learning models using Python libraries such as TensorFlow and Keras for real-world applications. (Apply) 7. Utilize deep learning techniques for image classification, object detection, sentiment analysis, and language modeling. (Apply)					
Course Content:						
odule 1	**sics of Neural Networks**	signment			**18[8L+10P] Sessions**	

pics:

Understanding Perceptron with Excel, Understanding Multilayer Perceptron with Excel, From Multilayer Perceptron to Deep Learning, Error Backpropagation and Gradient Descent to reduce errors, Activation Functions, Deep Learning, Problems with Deep Learning with solutions.

odule 2	**TensorFlow Basics**	signment		**14[7L+7P] Sessions**

pics:

Introduction to TensorFlow, TensorFlow dataset, Machine Learning with TensorFlow

odule 3	**Deep Learning methods with Tensor Flow and Keras**	signment		**14[6L+8P] Sessions**

Topics:

Main Features of TensorFlow, Keras basics, AI with Keras.

oject work/Assignment:

21. **Assignment 1 on (Module 1 and Module 2)**
22. **Assignment 2 on (Module 3)**

List of Laboratory Tasks:

Lab 1: Working with Deep Learning Frameworks
Objective: Explore various Deep Learning Frameworks
Tasks: Identify deep learning frameworks (Keras, Tensorflow, Matplotlib, etc)
Activity: Practice with various methods available in DL Frameworks to develop a Model.

Lab 2: Build a Basic Artificial Neural Network
Objective: Create a ANN with DL frameworks.
Task: Identify suitable ANN Layers using Keras and Tensorflow.
Activity: Design a basic Artificial Neural Networks using Keras with TensorFlow (pima-indians-diabetes)

Lab 3: Build a MultiLayer Perceptron
Objective: Create a MLP for classification task.
Task: Identify suitable model for house price prediction.
Activity: Design a MLP for implementing classification and fine-tuning using House price.csv

Lab 4: Create a Tensor in TensorFlow using List or Numpy array.
Objective: To understand how to create a tensor in TensorFlow using a Python list or NumPy array
Task: Create a simple tensor using both a Python list and a NumPy array in TensorFlow.
Activity: Create a tensor using a Python list and Numpy array

Lab 5: Apply math operations on tensor using various mathematical functions.
Objective: To learn how to apply mathematical operations on tensors using various TensorFlow mathematical functions.
Task: Perform basic mathematical operations (addition, subtraction, multiplication, division) and advanced functions (square, square root, exponential) on tensors.
Activity: Perform basic math operations: Add, Subtract, Multiply, Divide and Apply advanced math functions: Square, Square root, Exponential.

Lab 6: Connecting two tensors in dataset.
Objective: Combine two tensors using concatenation and stacking operations in TensorFlow.
Task: Combine two tensors using concatenation and stacking operations in TensorFlow
Activity: Concatenate them along a specific axis and Stack them along a new axis.

Lab 7: Building dataset from a file stored in a local drive
Objective: To learn how to build a dataset in TensorFlow from a file stored in a local drive.
Task: Load a dataset from a CSV file stored on the local drive and process it using TensorFlow
Activity: Load the file using TensorFlow's tf.data API and Process the dataset (e.g., convert it into tensors)

Lab 8: Loading Dataset from TensorFlow.dataset Library
Objective: To learn how to load a dataset from the tensorflow_datasets library and use it in machine learning models.
Task: Load a dataset from TensorFlow Datasets (tfds), preprocess it, and display sample data
Activity: Load a dataset (e.g., MNIST, CIFAR-10, IMDB Reviews) and Split the dataset into training and testing sets.

Lab 9: Build a Convolutional Neural Network
Objective: Create a CNN model.
Task: Build CNN architecture for Dog-Cat classification problem.
Activity: Implement a Convolution Neural Network (CNN) for dog/cat classification problem using keras.

Lab 10: **Build a Time-Series Model**
Objective: Create a RNN and LSTM Model
Task: Build RNN/LSTM Model for predicting time series data.
Activity Train a sentiment analysis model on IMDB dataset, use RNN layers with LSTM/GRU notes.

REFERENCE MATERIALS:

TEXTBOOKS

3. François Chollet, "Deep Learning with Python", 2nd Edition, Manning Publications, 2022
4. Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", MIT Press, 2017.

REFERENCES

4. Amlan Chakrabarti Amit Kumar Das, Saptarsi Goswami, Pabitra Mitra , "Deep Learning", Pearson Publication, 2021.
5. David Foster, "Generative Deep Learning" O'Reilly Publishers, 2020.
6. John D Kellehar, "Deep Learning", MIT Press, 2020.

JOURNALS/MAGAZINES

1. IEEE Transactions on Neural Networks and Learning Systems

https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=5962385

2. IEEE Transactions on Pattern Analysis and Machine Intelligence

https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=34http://ijaerd.com/papers/special_papers/IT032.pdf

3. International Journal of Intelligent Systems https://onlinelibrary.wiley.com/journal/1098111x

SWAYAM/NPTEL/MOOCs:

4. Swayam Nptel – Deep Learning – IIT Ropar https://onlinecourses.nptel.ac.in/noc21_cs35/preview
5. Coursera – Neural Networks and Deep Learning Andrew Ng
6. Coursera - Neural Networks for Machine Learning by Geoffrey Hinton in Coursera

urse Code: UG COURSE: I3429	urse Title: Deep Learning for Computer Vision pe of Course: Discipline Elective - Theory & Integrated Laboratory	T~ P~C	2	0		
rsion No.	)					
urse Pre~ requisites	MAT1003 Applied Statistics, Knowledge of Python, Machine Learning, and Digital image processing					
ti~requisites	L					
urse Description	This course covers the fundamentals and advanced concepts of deep learning for computer vision applications. Students will explore convolutional neural networks (CNNs), object detection, image segmentation, and generative models. Hands-on lab experiments will reinforce theoretical concepts using frameworks like TensorFlow and PyTorch.					
urse Out Comes	**On successful completion of the course the students shall be able to:** 1. **Understand the Fundamentals of Deep Learning for Vision** Explain the core concepts of neural networks and deep learning architectures for image processing. Implement and optimize convolutional neural networks (CNNs) for classification tasks. 2. **Apply Object Detection and Image Segmentation Techniques** Implement and analyze state-of-the-art object detection algorithms such as YOLO, Faster R-CNN, and SSD. Develop and evaluate image segmentation models like U-Net and Mask R-CNN. 3. **Explore Advanced Deep Learning Techniques for Vision** Utilize Vision Transformers (ViTs) and attention mechanisms for image classification. Generate and manipulate images using Generative Adversarial Networks (GANs). 4. **Deploy and Optimize Deep Learning Models for Real-World Applications**					
urse Content:						
dule 1	Fundamentals of Deep Learning for Vision	signment		actical		No. of Classes:8
Introduction to Deep Learning & Neural Networks, Convolutional Neural Networks (CNNs) Architecture Backpropagation & Optimization in CNNs, Transfer Learning & Pretrained Models.						
dule 2	Object Detection & Image Segmentation	signment		actical		No. of Classes:14
Introduction to Object Detection (R-CNN, SSD, YOLO), Region Proposal Networks (Faster R-CNN) Semantic & Instance Segmentation (U-Net, Mask R-CNN), Real-time Object Detection Applications						
dule 3	vanced Topics in Vision	signment		actical		No. of Classes:8
Attention Mechanisms & Vision Transformers (ViTs), Generative Adversarial Networks (GANs) for Image						

Generation, Self-supervised Learning for Vision, Multi-modal Learning (CLIP, DALL·E)				
odule 4	plications & Deployment	signment	ctical	No. of Classes:8
Edge AI & Mobile Deployment (TensorFlow Lite, ONNX), Adversarial Attacks & Robustness in Vision Models, Explainability & Interpretability of Vision Models, Case Studies & Industry Applications				

b Experiments are to be conducted on the following topics:-

Lab Sheet 1:

Keras Sequential API model

1. Read in the data and explore
2. Define a Sequential API model
3. Define the hyperparameters and optimizer
4. Train the model and visualize the history
5. Testing

Keras Functional API model:

1. Define a Functional API model
2. Train the model and visualize the history

Lab Sheet 2:

Softmax regression with Keras

1. Read in the data and prepare
2. Define a Sequential API model
3. Define the hyperparameters and optimizer
4. Train the model and visualize the history
5. Testing

Lab Sheet 3:

Convolutional Neural Network with Keras (grayscale images)

1. Read in the data:
2. Visualize the data:
3. Prepare the data:
4. Define a CNN model:
5. Define the hyperparameters and optimizer:
6. Train the model and visualize the history:
7. Testing:

Lab Sheet 4:

Convolutional Neural Network with Keras (color images):

1. Read in the data:
2. Visualize the data:
3. Prepare the data:
4. Define a CNN model:
5. Define the hyperparameters and optimizer:
6. Train the model and visualize the history:
7. Testing:

Lab Sheet 5:

Time series and prediction:

1. Read in the data and explore:
2. Apply the exponential smoothing method and predict

Recurrent neural network (RNN):

1. Pre-processing:
2. Do the necessary definitions: (Hyper parameters, Model,
3. Train the model:
4. Predict the future:

Lab Sheet 6:

Document classification with LSTM network:

1. Read in the data:
2. Explore the data:
3. Data preprocessing:
4. Define the model:
5. Define the optimizer and compile:
6. Train the model and visualize the history:
7. Testing:

Lab Sheet 7:

Document classification with LSTM network (Binary):

1. Read in the data:
2. Explore the data:
3. Data preprocessing:
4. Define the model:
5. Define the optimizer and compile:
6. Train the model and visualize the history:
7. Testing:

Lab Sheet 8:

Document classification with LSTM + CNN network (Binary):

1. Read in the data:
2. Explore the data:
3. Data preprocessing:
4. Define the model:
5. Define the optimizer and compile:
6. Train the model and visualize the history:
7. Testing:

Lab Sheet 9:

Softmax regression to recognize the handswritten digits:

1. Download the MNIST data:
2. Take a look at the dataset:
3. Do the necessary definitions:
4. Training and Testing:

Multi-layer neural network to recognize the handswritten digits:

1. Download the MNIST data:
2. Take a look at the dataset:
3. Do the necessary definitions:

Training and Testing:

Lab Sheet 10:

Object Detection using YOLOv5

Lab Sheet 11:

Image Segmentation using U-Net
Custom Object Detection using Faster R-CNN

Lab Sheet 12:

Implementing Vision Transformers for Image Classification
Generating Images using GANs (DCGAN, StyleGAN)

(Group Project)

8. Object Detection and Recognition:
    a. Haar cascade object detection (e.g., face detection or object detection using pre-trained classifiers).
    b. Feature-based object detection using techniques like Speeded-Up Robust Features (SURF) or Scale-Invariant Feature Transform (SIFT).
    c. Deep learning-based object detection using Convolutional Neural Networks (CNNs) or You Only Look Once (YOLO) algorithm.
9. Optical Character Recognition (OCR):
    a. Preprocessing of text images (e.g., binarization, noise removal, or skew correction).
    b. Text localization using techniques like connected component analysis or Stroke Width Transform (SWT).

c. Character recognition using machine learning algorithms like Support Vector Machines (SVM) or Convolutional Neural Networks (CNNs).

10. Gesture Recognition:
    a. Hand segmentation using techniques like background subtraction or skin color detection.
    b. Feature extraction from hand regions (e.g., finger counting, hand shape descriptors).
    c. Classification of gestures using machine learning algorithms (e.g., k-Nearest Neighbors or Support Vector Machines).

Tools/Software Required :

1. OpenCV 4
2. Python 3.7
3. MATLAB

xt Books

1. "Deep Learning for Computer Vision Image Classification, Object Detection and Face Recognition in Python" **Jason Brownlee (2019)**
2. "Deep Learning for Computer Vision with python" **Adrian Rosebrock (2017)**

References

3. **Goodfellow, I., Bengio, Y., & Courville, A. (2016).** *Deep Learning.* MIT Press.
   A foundational book covering deep learning principles, including CNNs, optimization, and generative models.
4. **Raschka, S., & Mirjalili, V. (2022).** *Machine Learning with PyTorch and Scikit-Learn.* Packt Publishing.
   Covers practical deep learning techniques using PyTorch, including CNNs and transfer learning.
5. **Geron, A. (2022).** *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (3rd Edition).* O'Reilly Media.
   Provides hands-on implementations of deep learning for computer vision using TensorFlow and Keras.
6. **Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2021).** *Dive into Deep Learning.* Available online (https://d2l.ai).
   Open-access book covering CNNs, object detection, and advanced vision techniques with PyTorch and TensorFlow.
7. **Chollet, F. (2021).** *Deep Learning with Python (2nd Edition).* Manning Publications.
   Explains deep learning fundamentals and applications with Keras, including image classification and segmentation.
8. **Ballé, J., Laparra, V., & Simoncelli, E. P. (2017).** *Deep Learning for Computer Vision: A Brief Introduction.*
   A concise introduction to CNNs, object detection, and generative models.

Ittagalpura, Rajanukunte, Yelahanka, Bengaluru 560 119