PRESIDENCY UNIVERSITY

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

PROGRAMME REGULATIONS & CURRICULUM

2024-28

PRESIDENCY SCHOOL OF COMPUTER SCIENCE & ENGINEERING

BACHELOR OF TECHNOLOGY (B.TECH.)

COMPUTER SCIENCE AND TECHNOLOGY (BIG DATA)

PRESIDENCY UNIVERSITY

Presidency University Act, 2013 of the Karnataka Act No. 41 of 2013 | Established under Section 2(f) of UGC Act, 1956
Approved by AICTE, New Delhi

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Program Regulations and Curriculum

2024-2028

BACHELOR OF TECHNOLOGY (B.Tech.) in

COMPUTER SCIENCE AND TECHNOLOGY (BIG DATA)

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-24.10/CSE19/CBD/2024-28

Resolution No.10 of the 24th Meeting of the Academic Council held on 03rd August 2024, and ratified by the Board of Management in its 24th Meeting held on 05th August, 2024.

AUGUST-2024

Table of Contents

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 Presidency School of Computer Science and Engineering, committed to developing globally competent engineers, dedicated to developing cutting-edge technology to enhance the quality of life.

1.4 Mission of Presidency School of Computer Science and Engineering

- Cultivate a practice-driven environment with computing-based pedagogy, integrating theory and practice.
- Attract and nurture world-class faculty to excel in teaching and research in the realm of computing sciences.
- Establish state-of-the-art computing facilities for effective teaching and learning experiences.
- Promote interdisciplinary studies to nurture talent for global impact.
- Instill 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 Social Project Based 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 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 2024-2028.
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 2024-2028 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. *"Dean" means the Dean of the concerned School;*

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

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

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 and Engineering;*

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;

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 2024-2028 are subject to, and, pursuant to the Academic Regulations, 2021. These Program Regulations shall be applicable to the following ongoing Bachelor of Technology (B.Tech.) Degree Programs of 2024-2028 offered by the Presidency School of Computer Science and Engineering (PSCS):

B.Tech. Computer Science and Engineering

2. B. Tech. Computer Science and Technology (Big Data)

3. B. Tech. Computer Science and Engineering (Block Chain)

4. B. Tech. Computer Science and Technology (DevOps)

5. B. Tech. Computer Science and Engineering (Cyber Security)

6. B. Tech. Computer Science and Engineering (Internet of Things)

7. B. Tech. Computer Science and Engineering (Data Science)

8. B. Tech. Computer Science and Technology [Artificial Intelligence and Machine Learning]

9. B. Tech. Information Science and Technology [Artificial Intelligence and Data Science]

10. B. Tech. Computer Science and Information Technology

11. B. Tech. Computer Science and Engineering (Networks)

12. B. Tech. Computer Engineering

13. B. Tech. Information Science and Engineering [Artificial Intelligence and Robotics]

14. B. Tech. Computer Science and Engineering (Artificial Intelligence and Machine Learning)

15. B. Tech. Artificial Intelligance and Data Science

16. B.Tech. Robotics and Artificial Intelligance

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

5.2 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.

5.3 The effect of periodic amendments or changes in the Program Regulations, on the students admitted in earlier years, shall be dealt with appropriately and carefully, so as to ensure that those students are not subjected to any unfair situation whatsoever, although they are required to conform to these revised Program Regulations, without any undue favour or considerations

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 of 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 (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 success as Computer Science and Engineering with innovative skills,moral and ethical values.

PEO2. Engage in lifelong learning through research and professional development,

PEO3. Serve as a leader in the profession through consultancy, extension activities or entrepreneurship.

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:

PSO 01: Problem Analysis: Identify, formulate, research literature, and analyse complex engineering problems related to Software Engineering principles & practice, Programming, Big Data computing & analytics Substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PSO 02: Design/development of Solutions: Design solutions for complex engineering problems related to Software Engineering principles & practice, Programming, Big Data Computing & analytics 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.

PSO 03:Modern Tools 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 & practice, Programming, Big Data Computing & analytics with an understanding of the limitations.

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 Technology-Big Data) 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 Mechanical 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., 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 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 10.1.1,10.1.2,10.1.3

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 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. 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

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
		Marks	25	25	25	25	50	50	-	200	
3	1-0-4-3	Percentage	-	25%	5%	20%	10%	40%	-	100%	Mid-Term & End Term by School
		Marks	-	25	5	20	10	40	-	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 in School level
		Marks	-	50	-	-	-	-	50	100	
6	0-0-2-1	Percentage	-	100%	-	-	-	-	-	100%	Only CA in 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	

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 Sub–Clauses 8.9.1 and 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 13.3.2 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 Technology-Big Data) Program Structure (2024-2028) totalling 160 credits. Table 3 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: B.Tech. (Computer Science and Technology Big Data) 2024-2028: Summary of Mandatory Courses and Minimum Credit Contribution from various Baskets

Sl. No.	Baskets	Credit Contribution
1	Humanities and Social Sciences including Management Courses (HSMC)	10
2	Basic Science Courses (BSC)	19
3	Engineering Science Courses (ESC)	23
4	Professional Core Courses (PCC)	68
5	Professional Elective Courses (PEC)	18
6	Open Elective Courses (OEC)	06
7	Project Work (PRW)	16
8	Mandatory Courses (MAC)	0
	Total Credits	**160 (Minimum)**

In the entire Program, the practical and skill based course component contribute to an extent of approximately 60% out of the total credits of 160 for B.Tech. (Computer Science and Technology Big Data) 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.

PART C – CURRICULUM STRUCTURE

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.

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 Humanities and Social Sciences including Management Courses (HSMC)

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	DES1146	Introduction to Design Thinking	1	0	0	1
4	ENG2001/ FRLXXXX	Advanced English / Foreign Language courses	1	0	2	2
5	PPS1012	Enhancing Personality Through Soft Skills	0	0	2	1
6	MGTXXXX	Managerial Economics and Financial Analysis	3	0	0	3

		Total No. of Credits	**10**

Table 3.2 : List of Basic Science Courses (BSC)

S.No	Course Code	Course Name	L	T	P	C
1	MAT1001	Calculus and Linear Algebra	3	0	2	4
2	PHY1002	Optoelectronics and Device Physics	2	0	2	3
3	MAT1003	Applied Statistics	2	0	0	2
4	MAT2501	Integral Transforms and Partial Differential Equations	3	0	0	3
5	MAT2602	Numerical Computations	3	0	0	3
6	MAT2605	Discrete Mathematics	4	0	0	4
		Total No. of Credits				**19**

Table 3.3 : List of Engineering Science Courses (ESC)

S.No	Course Code	Course Name	L	T	P	C
3	MEC1006	Engineering Graphics	2	0	0	2
6	CSE1004	Problem Solving Using C	1	0	4	3
7	ECE2007	Digital Design	2	0	2	3
3	CIV1008	Basic Engineering Sciences	2	0	0	2
4	CSE1006	Problem Solving using JAVA	1	0	4	3
7	EEE1007	Basics of Electrical and Electronics Engineering	3	0	2	4
9	ECE2010	Innovative Projects Using Arduino	-	-	-	1
6	CSE1500	Computational Thinking Using Python	2	0	2	3
7	CSE2510	Competitive Programming and Problem Solving	0	0	4	2
		Total No. of Credits				**23**

Table 3.4 : List of Professional Core Courses (PCC)

S.No	Course Code	Course Name	L	T	P	C
1	CBD1700	Introduction to Big Data	3	0	0	3
2	CBD2000	Data Structures and Algorithms	3	0	0	3
3	CSE1510	Database Management Systems	3	0	0	3
4	CSE1511	Database Management Systems Lab	0	0	2	1
5	CBD2001	Data Structures and algorithms Lab	0	0	4	2
6	CBD2500	Data Visualization and Reporting	3	0	0	3
7	CBD2501	Data Visualization and Reporting Lab	0	0	4	2
8	CBD2502	Data Mining and Predictive Analytics	3	0	0	3
9	CBD2503	Data Mining and Predictive Analytics Lab	0	0	2	1
10	CBD2504	Fundamentals of Big Data Analytics	3	0	0	3
11	CBD2505	Fundamentals of Big Data Analytics lab	0	0	4	2
12	CSE2000	Software Design and Development	3	0	0	3
13	CBD2506	Cloud Computing for Big Data	3	0	0	3
14	CBD2507	Cloud Computing for Big Data Lab	0	0	2	1
15	CBD2508	Big Data Technologies	3	0	0	3
16	CBD2509	Big Data Technologies Lab	0	0	4	2
17	CBD2510	No SQL Databases	3	0	0	3
18	CBD2512	No SQL Databases Lab	0	0	4	2
19	CBD2513	Web Intelligence and Analytics	2	0	0	2
20	CBD2514	Web Intelligence and Analytics Lab	0	0	2	1
21	CSE1700	Essentials of AI	3	0	0	3
22	CSE1701	Essentials of AI Lab	0	0	4	2
23	CSE1704	Deep Learning Techniques	3	0	0	3
24	CSE1705	Deep Learning Techniques Lab	0	0	4	2
25	CBD2515	Bioinformatics and Genomic Data Analytics	3	0	0	3
26	CBD2516	Big Data in Supply Chain and Logistics	3	0	0	3

27	CBD2517	Big Data in Supply Chain and Logistics Lab	0	0	4	2
28	CBD2518	Data Security and Cryptography	3	0	0	3
29	CBD2519	Data Security and Cryptography Lab	0	0	2	1
		Total No. of Credits				**68**

Table 3.5 : List of course in Project Work basket (PRW)					
S.No	Course Name	L	T	P	C
1	Capstone Project	0	0	0	10
2	Internship	0	0	0	2
3	Mini Project	0	0	0	4
				Total No. of Credits	**16**

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 4-6 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 Mini Project

A student may opt to do a Mini Project for a period of 6-8 weeks in an Industry / Company or academic / research institution or the University Department(s) 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 Mini Project 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 Mini Project in an Industry / Company or academic / research institution of her / his choice subject to the above mentioned condition (Sub-Clause 18.2.1). 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 12-14 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 Specialisations / Stream Basket

Table 3.6 : Professional Electives Courses/Specialization Tracks – Minimum of 12 credits is to be earned by the student in a particular track and overall 18 credits.						
Track 1 - Big Data with Cloud Computing						
S.No	Course Code	Course Name	L	T	P	C
1	CBD3400	Fundamentals of Cloud Computing	3	0	0	3
2	CBD3401	Distributed Computing with Hadoop	3	0	0	3
3	CBD3402	Edge Computing & IoT Integration with Cloud	3	0	0	3
4	CBD3403	Cloud Storage & Data Management	3	0	0	3
5	CBD3404	Cloud-Based Big Data Architecture & Optimization	3	0	0	3
6	CBD3405	Serverless Computing & Microservices in Cloud	3	0	0	3
Track 2 - Big Data in Data Science						
1	CBD3406	Introduction to Data Science & Big Data	3	0	0	3
2	CBD3407	Feature Engineering & Model Selection	3	0	0	3
3	CBD3408	Big Data-driven Business Intelligence	3	0	0	3
4	CBD3409	Time Series Analysis & Forecasting	3	0	0	3
5	CBD3410	Natural Language Processing for Big Data	3	0	0	3
6	CBD3411	Data Governance, Ethics & Privacy	3	0	0	3
Track 3 - Big Data with Artificial Intelligence						
1	CBD3412	Computer Vision for AI Applications	3	0	0	3
2	CBD3413	Reinforcement Learning for Big Data	3	0	0	3
3	CBD3414	Generative AI and Large Language Models (LLMs)	3	0	0	3

4	CBD3415	Explainable AI (XAI) and Ethical AI	3	0	0	3
5	CBD3416	Advanced Deep Learning Architectures	3	0	0	3
6	CBD3417	Real-Time Big Data Processing and AI Deployment	3	0	0	3

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

Table 3.7 : Open Elective Courses Baskets: Minimum Credits to be earned from this Basket is 12

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
Chemistry Basket											
1	CHE1003	Fundamentals of Sensors	3	0	0	3	S	ES	-	-	-
2	CHE1004	Smart materials for IOT	3	0	0	3	S	ES	-	-	-
3	CHE1005	Computational Chemistry	2	0	0	2	S	ES	-	-	-
4	CHE1006	Introduction to Nano technology	3	0	0	3	S	ES	-	-	-
5	CHE1007	Biodegradable electronics	2	0	0	2	S	ES	-	-	-
6	CHE1008	Energy and Sustainability	2	0	0	2	S	ES	-	-	-
7	CHE1009	3D printing with Polymers	2	0	0	2	S	ES	-	-	-
8	CHE1010	Bioinformatics and Healthcare IT	2	0	0	2	S	ES	-	-	-
9	CHE1011	Chemical and Petrochemical catalysts	3	0	0	3	S	ES	-	-	-
10	CHE1012	Introduction to Composite materials	2	0	0	2	S	ES	-	-	-
11	CHE1013	Chemistry for Engineers	3	0	0	3	S	ES	-	-	-
12	CHE1014	Surface and Coatings technology	3	0	0	3	S	ES	-	-	-
13	CHE1015	Waste to Fuels	2	0	0	2	S	ES	-	-	-
14	CHE1016	Forensic Science	3	0	0	3	S	ES	-	-	-
Civil Engineering Basket											
1	CIV1001	Disaster mitigation and management	3	0	0	3	S	-	-	-	-
2	CIV1002	Environment Science and Disaster Management	3	0	0	3	FC	-	-	-	-
3	CIV2001	Sustainability Concepts in Engineering	3	0	0	3	S	-	-	-	-
4	CIV2002	Occupational Health and Safety	3	0	0	3	S	-	-	-	-
5	CIV2003	Sustainable Materials and Green Buildings	3	0	0	3	EM	-	-	-	-
6	CIV2004	Integrated Project Management	3	0	0	3	EN	-	-	-	-

7	CIV2005	Environmental Impact Assessment	3	0	0	3	EN	-	-	-	-
8	CIV2006	Infrastructure Systems for Smart Cities	3	0	0	3	EN	-	-	-	-
9	CIV2044	Geospatial Applications for Engineers	2	0	2	3	EM	-	-	-	-
10	CIV2045	Environmental Meteorology	3	0	0	3	S	-	-	-	-
11	CIV3046	Project Problem Based Learning	3	0	0	3	S	-	-	-	-
12	CIV3059	Sustainability for Professional Practice	3	0	0	3	EN	-	-	-	-
Commerce Basket											
1	COM2001	Introduction to Human Resource Management	2	0	0	2	F	HP/GS	-	-	-
2	COM2002	Finance for Non Finance	2	0	0	2	S	-	-	-	-
3	COM2003	Contemporary Management	2	0	0	2	F	-	-	-	-
4	COM2004	Introduction to Banking	2	0	0	2	F	-	-	-	-
5	COM2005	Introduction to Insurance	2	0	0	2	F	-	-	-	-
6	COM2006	Fundamentals of Management	2	0	0	2	F	-	-	-	-
7	COM2007	Basics of Accounting	3	0	0	3	F	-	-	-	-
Computer Science Basket											
1	CSE2002	Programming in Java	2	0	2	3	S/EM	-	-	-	-
2	CSE2003	Social Network Analytics	3	0	0	3	S	GS	-	-	-
3	CSE2004	Python Application Programming	2	0	2	3	S/ EM	-	-	-	-
4	CSE2005	Web design fundamentals	2	0	2	3	S/ EM/EN	-	-	-	-
5	CSE3111	Artificial Intelligence : Search Methods For Problem Solving	3	0	0	3	S/ EM/EN	-	-	-	-
6	CSE3112	Privacy And Security In Online Social Media	3	0	0	3	S/ EM/EN	-	-	-	-
7	CSE3113	Computational Complexity	3	0	0	3	S/ EM/EN	-	-	-	-
8	CSE3114	Deep Learning for Computer Vision	3	0	0	3	S/ EM/EN	-	-	-	-
9	CSE3115	Learning Analytics Tools	3	0	0	3	S/ EM/EN	-	-	-	-
Design Basket											
1	DES1001	Sketching and Painting	0	0	2	1	S	-	-	-	-
2	DES1002	Innovation and Creativity	2	0	0	2	F	-	-	-	-
3	DES1121	Introduction to UX design	1	0	2	2	S	-	-	-	-
4	DES1122	Introduction to Jewellery Making	1	0	2	2	S	-	-	-	-
5	DES1124	Spatial Stories	1	0	2	2	S	-	-	-	-
6	DES1125	Polymer Clay	1	0	2	2	S	-	-	-	-
7	DES2001	Design Thinking	3	0	0	3	S	-	-	-	-
8	DES1003	Servicability of Fashion Products	1	0	2	2	F	ES	-	-	-
9	DES1004	Choices in Virtual Fashion	1	0	2	2	F	ES, GS, HP	-	-	-
10	DES1005	Fashion Lifestyle and Product Diversity	1	0	2	2	F	ES, GS, HP	-	-	-
11	DES1006	Colour in Everyday Life	1	0	2	2	F	ES	-	-	-
12	DES2080	Art of Design Language	3	0	0	3	S	-	-	-	-
13	DES2081	Brand Building in Design	3	0	0	3	S	-	-	-	-
14	DES2085	Web Design Techniques	3	0	0	3	S	-	-	-	-

15	DES2089	3D Modeling for Professionals	1	0	4	3	S	-	-	-	-
16	DES2090	Creative Thinking for Professionals	3	0	0	3	S	-	-	-	-
17	DES2091	Idea Formulation	3	0	0	3	S	-	-	-	-
Electrical and Electronics Basket											
1	EEE1002	IoT based Smart Building Technology	3	0	0	3	S	-	-	-	-
2	EEE1003	Basic Circuit Analysis	3	0	0	3	S	-	-	-	-
3	EEE1004	Fundamentals of Industrial Automation	3	0	0	3	S	-	-	-	-
4	EEE1005	Electric Vehicles & Battery Technology	3	0	0	3	S	-	-	-	-
5	EEE1006	Smart Sensors for Engineering Applications	3	0	0	3	S	-	-	-	-
Electronics and Communication Basket											
1	ECE1003	Fundamentals of Electronics	3	0	0	3	F	-	-	-	-
2	ECE1004	Microprocessor based systems	3	0	0	3	F	-	-	-	-
3	ECE3089	Artificial Neural Networks	3	0	0	3	S	-	-	-	-
4	ECE3097	Smart Electronics in Agriculture	3	0	0	3	F/EM	-	-	-	-
5	ECE3098	Environment Monitoring Systems	3	0	0	3	F/EM	-	-	-	-
6	ECE3102	Consumer Electronics	3	0	0	3	F/EM	-	-	-	-
7	ECE3103	Product Design of Electronic Equipment	3	0	0	3	S/F/ EM / EN	-	-	-	-
8	ECE3106	Introduction to Data Analytics	3	0	0	3	F/EM	-	-	-	-
9	ECE3107	Machine Vision for Robotics	3	0	0	3	F/EM	-	-	-	-
English Basket											
1	ENG1008	Indian Literature	2	0	0	2	-	GS/ HP	-	-	-
2	ENG1009	Reading Advertisement	3	0	0	3	S	-	-	-	-
3	ENG1010	Verbal Aptitude for Placement	2	0	2	3	S	-	-	-	-
4	ENG1011	English for Career Development	3	0	0	3	S	-	-	-	-
5	ENG1012	Gender and Society in India	2	0	0	2	-	GS/ HP	-	-	-
6	ENG1013	Indian English Drama	3	0	0	3	-	-	-	-	-
7	ENG1014	Logic and Art of Negotiation	2	0	2	3	-	-	-	-	-
8	ENG1015	Professional Communication Skills for Engineers	1	0	0	1	-	-	-	-	-
DSA Basket											
1	DSA2001	Spirituality for Health	2	0	0	2	F	HP	-	-	-
2	DSA2002	Yoga for Health	2	0	0	2	S	HP	-	-	-
3	DSA2003	Stress Management and Well Being	2	0	0	2	F	-	-	-	-
Kannada Basket											
1	KAN1001	Kali Kannada	1	0	0	1	S	-	-	-	-
2	KAN1003	Kannada Kaipidi	3	0	0	3	S	-	-	-	-
3	KAN2001	Thili Kannada	1	0	0	1	S	-	-	-	-
4	KAN2003	Pradharshana Kale	1	0	2	2	S	-	-	-	-
5	KAN2004	Sahithya Vimarshe	2	0	0	2	S	-	-	-	-
6	KAN2005	Anuvadha Kala Sahithya	3	0	0	3	S	-	-	-	-
7	KAN2006	Vichara Manthana	3	0	0	3	S	-	-	-	-
8	KAN2007	Katha Sahithya Sampada	3	0	0	3	S	-	-	-	-
9	KAN2008	Ranga Pradarshana Kala	3	0	0	3	S	-	-	-	-
Foreign Language Basket											
1	FRL1004	Introduction of French Language	2	0	0	2	S	S	-	-	-

2	FRL1005	Fundamentals of French	2	0	0	2	S	S	-	-	-
3	FRL1009	Mandarin Chinese for Beginners	3	0	0	3	S	S	-	-	-
Law Basket											
1	LAW1001	Introduction to Sociology	2	0	0	0	2	F	HP	-	-
2	LAW2001	Indian Heritage and Culture	2	0	0	0	2	F	HP/GS	-	-
3	LAW2002	Introdcution to Law of Succession	2	0	0	0	2	F	HP/GS	-	-
4	LAW2003	Introduction to Company Law	2	0	0	0	2	F	HP	-	-
5	LAW2004	Introduction to Contracts	2	0	0	2	F	HP	-	-	-
6	LAW2005	Introduction to Copy Rights Law	2	0	0	2	F	HP	-	-	-
7	LAW2006	Introduction to Criminal Law	2	0	0	2	F	HP	-	-	-
8	LAW2007	Introduction to Insurance Law	2	0	0	2	F	HP	-	-	-
9	LAW2008	Introduction to Labour Law	2	0	0	2	F	HP	-	-	-
10	LAW2009	Introduction to Law of Marriages	2	0	0	2	F	HP/GS	-	-	-
11	LAW2010	Introduction to Patent Law	2	0	0	2	F	HP	-	-	-
12	LAW2011	Introduction to Personal Income Tax	2	0	0	2	F	HP	-	-	-
13	LAW2012	Introduction to Real Estate Law	2	0	0	2	F	HP	-	-	-
14	LAW2013	Introduction to Trademark Law	2	0	0	2	F	HP	-	-	-
15	LAW2014	Introduction to Competition Law	3	0	0	3	F	HP	-	-	-
16	LAW2015	Cyber Law	3	0	0	3	F	HP	-	-	-
17	LAW2016	Law on Sexual Harrassment	2	0	0	2	F	HP/GS	-	-	-
18	LAW2017	Media Laws and Ethics	2	0	0	2	F	HP/GS	-	-	-
Mathematics Basket											
1	MAT2008	Mathematical Reasoning	3	0	0	3	S	-	-	-	-
2	MAT2014	Advanced Business Mathematics	3	0	0	3	S	-	-	-	-
3	MAT2041	Functions of Complex Variables	3	0	0	3	S	-	-	-	-
4	MAT2042	Probability and Random Processes	3	0	0	3	S	-	-	-	-
5	MAT2043	Elements of Number Theory	3	0	0	3	S	-	-	-	-
6	MAT2044	Mathematical Modelling and Applications	3	0	0	3	S	-	-	-	-
Mechanical Basket (not to be offered for Mechanical Department students)											
1	MEC1001	Fundamentals of Automobile Engineering	3	0	0	3	F	-	-	-	-
2	MEC1002	Introduction to Matlab and Simulink	3	0	0	3	S/EM	-	-	-	-
3	MEC1003	Engineering Drawing	1	0	4	3	S	-	-	-	-
4	MEC2001	Renewable Energy Systems	3	0	0	3	F	ES	-	-	-
5	MEC2002	Operations Research & Management	3	0	0	3	F	-	-	-	-
6	MEC2003	Supply Chain Management	3	0	0	3	S/ EM/ EN	-	-	-	-
7	MEC2004	Six Sigma for Professionals	3	0	0	3	S/EM	-	-	MEC 2008	-
8	MEC2005	Fundamentals of Aerospace Engineering	3	0	0	3	F	-	-	-	-
9	MEC2006	Safety Engineering	3	0	0	3	S/EM	ES	-	-	-
10	MEC2007	Additive Manufacturing	3	0	0	3	F/EM	-	-	-	-
11	MEC3069	Engineering Optimisation	3	0	0	3	S/EM	-	-	-	-

12	MEC3070	Electronics Waste Management	3	0	0	3	F/S	ES	-	-	-
13	MEC3071	Hybrid Electric Vehicle Design	3	0	0	3	S/EM	ES	-	-	-
14	MEC3072	Thermal Management of Electronic Appliances	3	0	0	3	S/EM	-	-	-	-
15	MEC3200	Sustainable Technologies and Practices	3	0	0	3	S/EM	-	-	-	-
16	MEC3201	Industry 4.0	3	0	0	3	S/EM	-	-	-	-
Petroleum Basket											
1	PET1011	Energy Industry Dynamics	3	0	0	3	FC	ES	-	NIL	-
2	PET1012	Energy Sustainability Practices	3	0	0	3	FC	ES	-	NIL	-
Physics Basket											
1	PHY1003	Mechanics and Physics of Materials	3	0	0	3	FC / SD				
2	PHY1004	Astronomy	3	0	0	3	FC				
3	PHY1005	Game Physics	2	0	2	3	FC / SD				
4	PHY1006	Statistical Mechanics	2	0	0	2	FC				
5	PHY1007	Physics of Nanomaterials	3	0	0	3	FC				
6	PHY1008	Adventures in nanoworld	2	0	0	2	FC				
7	PHY2001	Medical Physics	2	0	0	2	FC	ES			
8	PHY2002	Sensor Physics	1	0	2	2	FC / SD				
9	PHY2003	Computational Physics	1	0	2	2	FC				
10	PHY2004	Laser Physics	3	0	0	3	FC	ES			
11	PHY2005	Science and Technology of Energy	3	0	0	3	FC	ES			
12	PHY2009	Essentials of Physics	2	0	0	2	FC				
Management Basket- I											
1	MGT2007	Digital Entrepreneurship	3	0	0	3	S/EM/EN	-	-	-	-
2	MGT2015	Engineering Economics	3	0	0	3	S	-	-	-	-
3	MGT2023	People Management	3	0	0	3	S/EM/ EN	HP	-	-	-
Management Basket- II											
1	MGT1001	Introduction to Psychology	3	0	0	3	F	HP	-	-	-
2	MGT1002	Business Intelligence	3	0	0	3	EN	-	-	-	-
3	MGT1003	NGO Management	3	0	0	3	S	-	-	-	-
4	MGT1004	Essentials of Leadership	3	0	0	3	EM/ EN	GS/ HP	-	-	-
5	MGT1005	Cross Cultural Communication	3	0	0	3	S/EM/ EN	HP	-	-	-
6	MGT2001	Business Analytics	3	0	0	3	S/ EM/EN	-	-	-	-
7	MGT2002	Organizational Behaviour	3	0	0	3	F	HP	-	-	-
8	MGT2003	Competitive Intelligence	3	0	0	3	S	-	-	-	-
9	MGT2004	Development of Enterprises	3	0	0	3	S/EM/EN	-	-	-	-
10	MGT2005	Economics and Cost Estimation	3	0	0	3	S/EM	-	-	-	-
11	MGT2006	Decision Making Under Uncertainty	3	0	0	3	S	-	-	-	-
12	MGT2008	Econometrics for Managers	3	0	0	3	S	-	-	-	-
13	MGT2009	Management Consulting	3	0	0	3	S/EM/EN	-	-	-	-
14	MGT2010	Managing People and Performance	3	0	0	3	S/EM/EN	HP/GS	-	-	-
15	MGT2011	Personal Finance	3	0	0	3	F	-	-	-	-
16	MGT2012	E Business for Management	3	0	0	3	S/EM	-	-	-	-

17	MGT2013	Project Management	3	0	0	3	EN / EM	GS/HP/ ES	-	-	-
18	MGT2014	Project Finance	3	0	0	3	EN / EM	HP	-	-	-
19	MGT2016	Business of Entertainment	3	0	0	3	EM/ EN	-	-	-	-
20	MGT2017	Principles of Management	3	0	0	3	S/EM/ EN	-	-	-	-
21	MGT2018	Professional and Business Ethics	3	0	0	3	S/EM/ EN	HP	-	-	-
22	MGT2019	Sales Techniques	3	0	0	3	S/EM/ EN	HP	-	-	-
23	MGT2020	Marketing for Engineers	3	0	0	3	S/EM/ EN	HP	-	-	-
24	MGT2021	Finance for Engineers	3	0	0	3	S/EM/ EN	HP	-	-	-
25	MGT2022	Customer Relationship Management	3	0	0	3	S/EM/ EN	HP	-	-	-
Media Studies Basket											
1	BAJ3050	Corporate Filmmaking and Film Business	0	0	4	2	EM	HP	-	-	-
2	BAJ3051	Digital Photography	2	0	2	3	EM	HP	-	-	-
3	BAJ3055	Introduction to News Anchoring and News Management	0	0	2	1	EM	-	-	-	-

21. List of NPTEL Courses for B.Tech. Compputer Science and Engineering- Block Chain

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

Sl. No.	Course Code	Course Name	L	T	P	Credits	Contact Hours	Basket
Semester 1 - Physics Cycle						19	26	
1	MAT1001	Calculus and Linear Algebra	3	0	2	4	5	BSC
2	PHY1002	Optoelectronics and Device Physics	2	0	2	3	4	BSC
3	MEC1006	Engineering Graphics	2	0	0	2	2	ESC
4	ENG1002	Technical English	1	0	2	2	3	HSMC
5	PPS1001	Introduction to soft skills	0	0	2	1	2	HSMC
6	CSE1004	Problem Solving Using C	1	0	4	3	5	ESC
7	ECE2007	Digital Design	2	0	2	3	4	ESC
8	DES1146	Introduction to Design Thinking	1	0	0	1	1	HSMC
Semester 2 - BES Cycle						15	23	

1	MAT1003	Applied Statistics	2	0	0	2	2	BSC
2	CHE1018	Environmental Science	1	0	2	0	3	MAC
3	CIV1008	Basic Engineering Sciences	2	0	0	2	2	ESC
4	CSE1006	Problem Solving using JAVA	1	0	4	3	5	ESC
5	ENG2001/ FRLXXXX	Advanced English / Foreign Language courses	1	0	2	2	3	HSMC
6	PPS1012	Enhancing Personality Through Soft Skills	0	0	2	1	2	HSMC
7	EEE1007	Basics of Electrical and Electronics Engineering	3	0	2	4	5	ESC
8	LAW1007	Indian Constitution and Professional Ethics for Engineers	1	0	0	0	1	MAC
9	ECE2010	Innovative Projects Using Arduino	-	-	-	1	0	ESC
Semester 3						25	27	
1	MAT2501	Integral Transforms and Partial Differential Equations	3	0	0	3	3	BSC
2	CBD1700	Introduction to Big Data	3	0	0	3	3	PCC
3	CBD2000	Data Structures and Algorithms	3	0	0	3	3	PCC
4	MAT2605	Discrete Mathematics	4	0	0	4	4	BSC
5	CSE1500	Computational Thinking Using Python	2	0	2	3	2	ESC
6	CSE1510	Database Management Systems	3	0	0	3	3	PCC
7	MGTXXXX	Managerial Economics and Financial Analysis	3	0	0	3	3	HSMC
8	CSE1511	Database Management Systems Lab	0	0	2	1	2	PCC
9	CBD2001	Data Structures and algorithms Lab	0	0	4	2	4	PCC
Semester 4						24	28	
1	MAT2602	Numerical Computations	3	0	0	3	3	BSC
2	CBD2500	Data Visualization and Reporting	3	0	0	3	3	PCC
3	CBD2501	Data Visualization and Reporting Lab	0	0	4	2	2	PCC
4	CBD2502	Data Mining and Predictive Analytics	3	0	0	3	3	PCC
5	CBD2503	Data Mining and Predictive Analytics Lab	0	0	2	1	2	PCC
6	CBD2504	Fundamentals of Big Data Analytics	3	0	0	3	3	PCC
7	CBD2505	Fundamentals of Big Data Analytics lab	0	0	4	2	4	PCC
8	CSE2000	Software Design and Development	3	0	0	3	3	PCC
9	CBD2506	Cloud Computing for Big Data	3	0	0	3	3	PCC
10	CBD2507	Cloud Computing for Big Data Lab	0	0	2	1	2	PCC
Semester 5						28	29	
1	CBD2508	Big Data Technologies	3	0	0	3	3	PCC
2	CBD2509	Big Data Technologies Laboratory	0	0	4	2	2	PCC

3	CBD2510	No SQL Databases	3	0	0	3	3	PCC
4	CBD2512	No SQL Databases Lab	0	0	4	2	2	PCC
5	CBD2513	Web Intelligence and Analytics	2	0	0	2	2	PCC
6	CBD2514	Web Intelligence and Analytics Laboratory	0	0	2	1	2	PCC
7	CSE1700	Essentials of AI	3	0	0	3	3	PCC
8	CSE1701	Essentials of AI Lab	0	0	4	2	4	PCC
9	CBDXXXX	Professional Elective – I	3	0	0	3	3	PEC
10	CSE1704	Deep Learning Techniques	3	0	0	3	3	PCC
11	CSE1705	Deep Learning Techniques Lab	0	0	4	2	2	PCC
12	CSE7000	Internship	-	-	-	2	0	PRW
Semester 6						23	30	
1	CBD2515	Bioinformatics and Genomic Data Analytics	3	0	0	3	3	PCC
2	CBD2516	Big Data in Supply Chain and Logistics	3	0	0	3	3	PCC
3	CBD2517	Big Data in Supply Chain and Logistics Lab	0	0	4	2	4	PCC
4	CBD2518	Data Security and Cryptography	3	0	0	3	3	PCC
5	CBD2519	Data Security and Cryptography Lab	0	0	2	1	2	PCC
6	CBDXXXX	Professional Elective – II	3	0	0	3	3	PEC
7	CBDXXXX	Professional Elective – III	3	0	0	3	3	PEC
8	PPSXXXX	Industry Preparedness Program	2	0	0	0	2	MAC
9		Open Elective – I	3	0	0	3	3	OEC
10	CSE2510	Competitive Programming and Problem Solving	0	0	4	2	4	ESC
Semester 7						16	12	
1	CBCXXXX	Professional Elective – IV	3	0	0	3	3	PEC
2	CBCXXXX	Professional Elective – V	3	0	0	3	3	PEC
3	CBCXXXX	Professional Elective – VI	3	0	0	3	3	PEC
4	XXXXXXX	Open Elective – II	3	0	0	3	3	OEC
5	CSE7100	Mini Project				4	0	PRW
Semester 8						10	0	
1	CSE7300	Capstone Project	-	-	-	10	0	PRW
						160		

Sample Catalogue is given below for reference:

IV. Course Catalogues:

Each course shall have a course catalogue with the following details:

i) Pre –Requisites of the course

ii)Course Description

iii) Course Outcome

iv) Course Content

iv) Reference Resources.

The Course Catalogues for the Courses offered in each basket are attached below:

Course Code: MAT1001	Course Title: Calculus and Linear Algebra Type of Course: Basic Sciences Theory	L-T- P-C	3	0	2	4
Version No.	3.0					
Course Pre-requisites	Basic Concepts of Limits, Differentiation, Integration					
Anti-requisites	NIL					
Course 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.					
Course Objective	The objective of the course is Skill Development of student by using Problem Solving Techniques.					
Course 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.			
Course Content:				
Module 1	Linear Algebra			10 Sessions
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: Eigenvalues 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. Engineering Applications of Linear Algebra.				
Module 2	Partial Derivatives			10 Sessions
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. Engineering Applications of partial derivatives.				
Module 3	Advanced Integral calculus			12 Sessions

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.

Engineering applications of partial derivatives.

Module 4	Ordinary Differential Equations	Assignment	Programming	12 Sessions

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 eax, sinax, cosax, eaxf(x), xnf(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.

Engineering applications of differential equations.

List of Laboratory Tasks:

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

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

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

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

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

Experiment No. 5 Computation of Area under a curve.

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

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

Experiment No. 8 Solution of Partial Differential equation

Experiment No. 9 solution using Cauchy Equation and Lagrange's Equation

Targeted Application & Tools that can be used:

The contents of this course has direct applications in most of the core engineering courses for problem formulations, Problem Solution and system Design.

Tools Used: MatLab, Zylink.

Assignment:

List at least 3 sets of Matrix Applications concerning the respective branch of Engineering and obtain the solution using MATLAB.

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.

Text Book

Sankara Rao, Introduction to Partial differential equations, Prentice Hall of India, edition, 2011

B. S. Grewal (2017), Higher Engineering Mathematics by, 44th Edition, Khanna Publishers.

References:

Victor Henner, Tatyana Belozerova, Mickhail Khenner, Ordinary and Partial Differential Equations, CRC Press, Edition, 2013.

Walter Ledermann, Multiple integrals, Springer, 1st edition

Lay, Linear Algebra ansd its applications, 3rd Ed., 2002, Pearson Education India.

Erwin Kreyzig, Advanced Engineering Mathematics, John Wiley and sons, Inc.10th Edition

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/

Topics relevant to the development of Foundation Skills: All solution methods

Topics relevant to development of Employability skills: Use of Matlab software.

Course Code: PHY1002	Course Title: Optoelectronics and Device Physics Type of Course: Engineering Sciences Theory	L-T-P-C	2-0-2-3
Version No.	1.0		
Course Pre-requisites	NIL		
Anti-requisites	NIL		
Course 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.		
Course 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].		

Course 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			
Course Content:				
Module 1	Fundamentals of Materials.	Assignment	Plotting of magnetization (M) v/s Magnetic field (H) for diamagnetic, paramagnetic and ferromagnetic materials using excel/ origin software.	7 Sessions
Topics: Concept of energy bands, charge carriers, carrier concentration, concept of Fermi level, Hall effect, Magnetic materials, Superconductors:				
Module 2	Advanced Devices and applications	Assignment	Data collection on efficiency of solar cells.	8 Sessions
Topics: p-n junctions, Zener diode, transistor characteristics, Optoelectronic devices:, Solar cells, I-V characteristics, and LEDs				
Module 3	Quantum concepts and Applications	Term paper	Seminar on quantum computers.	8 Sessions
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				
Module 4	Lasers and Optical fibers	Term paper	Case study on medical applications of Lasers.	7 Sessions

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.

List of Laboratory Tasks:

Experiment No. 1: Experimental errors and uncertainty using excel

Level 1: Calculation of accuracy and precision of a given data

Level 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.

Level 1: Determination of Wavelength of Laser

Level 2: Finding the particle size of lycopodium powder.

Experiment No. 3: To determine the proportionality of Hall Voltage, magnetic flux density and the polarity of Charge carrier.

Level 1: To determine the proportionality of Hall Voltage and magnetic flux density

Level 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.

Level 1: To study I –V characteristics of the given Zener diode in reverse bias and to determine break down voltage.

Level 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.

Level 1: To determine the input resistance of a given transistor.

Level 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.

Level 1 To study the current vs voltage characteristics of CdS photo-resistor at constant irradiance.

Level 2: To measure the photo-current as a function of the irradiance at constant voltage.

Experiment No. 8: To study the I-V characteristics and I-R characteristics of a solar cell as a function of the irradiance.

Level 1: To study the I-V characteristics

Level 2: I-R characteristics of a solar cell as a function of the irradiance.

Experiment No. 9: Calculate the numerical aperture and study the losses that occur in optical fiber cable. .

Level 1: Calculate the numerical aperture.

Level 2: study the losses that occur in optical fiber cable.

Experiment No. 10: To determine the magnetic susceptibility of a given diamagnetic and paramagnetic substances using Quincke's method.

Level 1: To determine the magnetic susceptibility of a given diamagnetic substance.

Level 2: To determine the magnetic susceptibility of a given paramagnetic substance.

Experiment No. 11: Plotting I-V characteristics in forward and reverse bias for LEDs and Determination of knee voltage.

Level 1: Plotting I-V characteristics in forward and reverse bias for LEDs

Level 2: Determination of knee voltage.

Experiment No. 12: Determination of Stefan's constant and verification of Stefan-Boltzmann Law.

Level 1: Determination of Stefan's constant

Level 2: Verification of Stefan-Boltzmann Law.

Targeted Application & Tools that can be used:

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.

Origin, excel and Mat lab soft wares for programming and data analysis.

Project 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.

Text Book

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:

https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=553045&site=ehost-live

https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=833068&site=ehost-live

https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=323988&site=ehost-live

https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=1530910&site=ehost-live

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: MEC1006	Course Title: Engineering Graphics Type of Course: Engineering Science & 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.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of "Engineering Graphics" and attain SKILL DEVELOPMENT through Problem solving methodologies.					
Course Outcomes	On successful completion of this course the students shall be able to: Demonstrate competency of Engineering Graphics as per BIS conventions and standards. Comprehend the theory of projection for drawing projections of Points, Lines and Planes under different conditions. Prepare multiview orthographic projections of Solids by visualizing them in different positions. Prepare pictorial drawings using the principles of isometric projections to visualize objects in three dimensions.					
Course Content:						
Module 1	Introduction to Drawing	Assignment	Standard technical drawing		02 Sessions	
Topics:						

Introduction, drawing instruments and their uses, relevant BIS conventions and standards, Lettering, Line conventions, dimensioning, Selection of drawing sheet size and scale. [02 Hours: Comprehension Level]				
Module 2	Orthographic projections of Points, Straight Lines and Plane Surfaces	Assignment	Projection methods Analysis	10 Sessions
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 in all 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]				
Module 3	Orthographic Projections of Solids	Assignment	Multi-view drawing Analysis	10 Sessions
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: K.R. Gopalakrishna, “Engineering Graphics”, Subhash Publishers, Bangalore. D. M. Kulkarni, A. P. Rastogi, A. K. Sarkar, “Engineering Graphics with AutoCAD,” Prentice Hall. 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: ENG1002	Course Title: Technical English Type of Course: Humanities Science / Theory	L-T-P-C	1-0-2-2
Version No.	V. 3		
Course Pre-requisites	Intermediate Level English		
Course	NIL		

Anti-requisites				
Course 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.			
Course 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: Develop proficiency in using technical vocabulary and terminology. Apply language skills for better speaking skills in technical fields. Write technical descriptions 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	9 Classes
Introduction to Technical English Differences between Technical English and General English Technical Writing Basics Technical Vocabulary				
Module 2	Technical Presentation	Presentations	Speaking Skills	12 Classes
Introduction Planning the Presentation Creating the Presentation Giving the Presentation				

Module 3	Technical Description	Assignment	Group Presentation	12 Classes
Product Description Process Description User Manuals Transcoding: Diagrams, charts and images				
Module 4	Technical Writing	Assignment	Writing Skills	12 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				
List of Laboratory Tasks: Module-1 Level 1: Worksheets Level 2: Worksheets Module 2				

Level 1: Preparing Presentation Level 2: Giving Presentation (Individual) Module-3 Level 1: Product Description & User Manual Level 2: Process Description & Transcoding Module 4 Level 1: Email Writing Level 2: Report Writing
Targeted Applications & Tools that can be used: Flipgrid Quizzes Youtube Videos Podcast
Project 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.
The following individual, as well as group Assignments, will be given to the students. Presentation Describing a product/process Individual Reports
Text Books Kumar, Sanjay; Pushpalatha. English Language and Communication Skills for Engineers. Oxford University Press. 2018. Brieger, Nick and Alison Paul. Technical English Vocabulary and Grammar.

https://nmetau.edu.ua/file/technical_english_vocabulary_and_grammar.pdf
Reference 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 Comfort, 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)
Topics Relevant to the Development of Employability Skills: Speaking Skills, Writing Skills, Critical Thinking and Critical Analysis, and Group Communication.

Course Code: PPS 1001	Course Title: Introduction to Soft Skills Type of Course: Practical Only Course	L- T- P- C	0	2	1
Version No.	1.0				

Course Pre-requisites	Students are expected to understand Basic English. Students should have desire and enthusiasm to involve, participate and learn.		
Anti-requisites	NIL		
Course 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.		
Course 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.		
Course 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		
Course Content:			
Module 1	INTRODUCTION TO SOFT SKILLS	Classroom activity	04 Hours
Topics: Setting Expectations, Ice Breaker, Significance of soft skills, Formal grooming, punctuality			
Module 2	EFFECTIVE COMMUNICATION	Individual 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.			
Module 3	HABIT FORMATION	Worksheets & Assignment	4 Hours
Topics: Professional and personal ethics for success, Identity based habits, Domino effect, Habit Loop, Unlearning, standing up for what is right			
Module 4	Goal setting & Time Management	Goal 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			
Individual Assessment 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. ACAlso 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: Write algorithms and to draw flowcharts for solving problems Demonstrate knowledge and develop simple applications in C programming constructs Develop and implement applications using arrays and strings Decompose a problem into functions and develop modular reusable code Solve applications in C using structures and Union 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) Programs using IO Statements, Conditional Statements and Looping Statements 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): Yashwant Kanetkar, Let us C, 17th Edition, BPB Publications, 2020. ReemaThareja, “Programming in C”, Oxford University Press, Second Edition, 2016. Kernighan, B.W and Ritchie,D.M, “The C Programming language”, Second Edition, Pearson Education, 2015 Schildt Herbert, “C: The Complete Reference”, Tata McGraw Hill Education, 4th Edition, 2014. 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: ECE2007	Course Title: Digital Design Type of Course: Theory &Integrated Laboratory	L- T-P-C	2	0	2	3
Version No.	2.0					
Course Pre-requisites	[1] Elements of Electronics/Electrical Engineering, 2] Basic concepts of number representation, Boolean Algebra					
Anti-requisites	NIL					
Course 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.					
Course 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.					
Course Outcomes	On successful completion of this course the students shall be able to: Describe the concepts of number systems, Boolean algebra and logic gates. Apply minimization techniques to simplify Boolean expressions. Demonstrate the Combinational circuits for a given logic Demonstrate the Sequential and programmable logic circuits Implement various combinational and sequential logic circuits using gates.					

<table>
<tr><td>Course Content:</td><td colspan="4"></td></tr>
<tr><td>Module 1</td><td>Fundamentals of Number systems- Boolean algebra and digital logic</td><td>Application Assignment</td><td>Data Analysis task</td><td>06 classes</td></tr>
<tr><td colspan="5">Topics:

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.</td></tr>
<tr><td>Module 2</td><td>Boolean function simplification</td><td>Application Assignment</td><td>Data Analysis task</td><td>08 Classes</td></tr>
<tr><td colspan="5">Topics:

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.</td></tr>
<tr><td>Module 3</td><td>Combinational Logic circuits:</td><td>Application Assignment</td><td>Programming Task & Data Analysis task</td><td>08 Classes</td></tr>
<tr><td colspan="5">Topics:

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.</td></tr>
<tr><td colspan="5">List of Laboratory Tasks:

Experiment N0 1: Verify the Logic Gates truth table

Level 1: By using Digital Logic Trainer kit

Level 2: By using Analog devices like RPS, Volt meter, Resistors and ICs</td></tr>
</table>

Experiment No. 2: Verify the Boolean Function and Rules

Level 1: By using Digital Logic Trainer kit

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

Experiment No. 3: Design and Implementations of HA/FA

Level 1: By using basic logic gates and Trainer Kit

Level 2: By using Universal logic gates and Trainer Kit

Experiment No. 4: Design and Implementations of HS/FS

Level 1: By using basic logic gates and Trainer Kit

Level 2: By using Universal logic gates and Trainer Kit

Experiment No. 5: Design and Implementations of combinational logic circuit for specifications

Level 1: Specifications given in the form of Truth table

Level 2: Specification should be extracted from the given scenario

Experiment No. 6: Study of Flip flops

Experiment No. 7: Design and Implementations of sequential logic circuit for specifications

Level 1: Specifications given in the form of Truth table

Level 2: Specification should be extracted from the given scenario

Experiment No.8: HDL coding for basic combinational logic circuits

Level 1: Gate level Modeling

Level 2: Behavioral Modeling

Experiment No.9: HDL coding for basic sequential logic circuit

Level 1: Gate level Modeling

Level 2: Behavioral Modeling

Targeted Application & Tools that can be used:

Digital 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

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

Text Book(s):

Mano, M. Morris and Ciletti Michael D., "Digital Design", Pearson Education, 6th edition

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)

eBook1: Mano, M. Morris and Ciletti Michael D., “Digital Design”, Pearson Education.

{[PDF] Digital Design By M. Morris Mano, Michael D Ciletti Book Free Download

}

eBook2:Floyd “DIGITAL LOGIC DESIGN” fourth edition- ePub, eBook- [PDF] DIGITAL LOGIC DESIGN FOURTH EDITION FLOYD | abri.engenderhealth.org.

NPTEL Course- NPTEL :: Electrical Engineering - NOC:Digital Electronic Circuits

Digital Logic Design PPT Slide 1 (iare.ac.in)

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:

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.

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)

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.

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.

Course Code: DES1146	Course Title: Introduction to Design Thinking Type of Course: Theory	L-T-P- C	1	0	0	1
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	The course aims to introduce students to the fundamental principles and processes of Design Thinking and will learn to apply Design Thinking methodologies to real-world challenges. The course emphasizes empathy, creativity, and collaboration, equipping students with essential skills for successful engineering practice.					
Course Objective	This course is designed to develop and familiarize the learners with the concepts of creating thinking and attain Entrepreneurship by using Participative Learning techniques.					
Course Outcomes	On successful completion of the course the students shall be able to: Understand the concept and importance of Design Thinking. Differentiate between traditional problem-solving and Design Thinking. Identify the core stages of the Design Thinking process.					
Course Content:	All assignments and projects must be developed using the reference materials available from the PU e-resource database – JSTOR, EBSCO, Library OPAC, NPTEL Videos, etc.					

Module 1	Introduction to Design Thinking	Visual journal, book of essays, context-specific assignment/project		Visual output generation, by Visual Journal and narrative development.	3 hours
Topic Definition and Introduction to Design Thinking Understand the Design Thinking Process					
Module 2	Design Thinking in Action	Visual journal, book of essays, context-specific assignment/project		Visual output generation, by visual journal and narrative development.	12 hours
Topics: Introduction to the steps of Design Thinking Process Understand use cases of Design thinking Design Thinking and Research Tools pertaining to Consumer Tech. , Home Tech. , Personal Tech. , Auto Tech. or Extended Reality.					
Targeted Application & Tools that can be used: Design ideation tools like Miro , SCAMPER etc. Research Tools for Human Centric Design using forecasting tools like WGSN Feedback tools like Google Forms , etc. Expert Lectures					
Text Book					

Thinking Design by S Balaram. New Delhi [India]: Sage Publications Pvt. Ltd. 2010. eBook., Database: eBook Collection (EBSCOhost)

https://puniversity.informaticsglobal.com:2284/ehost/detail/detail?vid=6&sid=18ab1f43-1f92-4d02-ae2e-a9c06dc06d8c%40redis&bdata=JnNpdGU9ZWhvc3QtbGl2ZQ%3d%3d#AN=354920&db=nlebk

References

Design Thinking by Clarke, Rachel Ivy. Series: Library Futures, Vol. 4. Chicago: ALA Neal-Schuman. 2020. eBook., Database: eBook Collection (EBSCOhost)

https://puniversity.informaticsglobal.com:2282/ehost/detail/detail?vid=4&sid=c80a7d79-eda4-4b7e-a0d6-afafe437962b%40redis&bdata=JnNpdGU9ZWhvc3QtbGl2ZQ%3d%3d#AN=2433506&db=nlebk

The Pocket Universal Methods of Design: 100 Ways to Research Complex Problems, Develop Innovative Ideas, and Design Effective Solutions by Bruce Hanington; Bella Martin. Minneapolis: Rockport Publishers. 2017. eBook., Database: eBook Collection (EBSCOhost)

https://puniversity.informaticsglobal.com:2282/ehost/detail/detail?vid=11&sid=f086b8c2-260e-4caa-8c48-d732c21a7724%40redis&bdata=JnNpdGU9ZWhvc3QtbGl2ZQ%3d%3d#AN=1638693&db=nlebk

What Is Design Thinking and Why Is It Important? By Rim Razzouk and Valerie Shute - Review of Educational Research, Vol. 82, No. 3 (September 2012), pp. 330-348 (19 pages), Published by: American Educational Research Association

https://puniversity.informaticsglobal.com:2054/stable/23260048?Search=yes&resultItemClick=true&searchText=design+thinking&searchUri=%2Faction%2FdoBasicSearch%3FQuery%3Ddesign%2Bthinking%26so%3Drel&ab_segments=0%2FSYC-6168%2Ftest&refreqid=fastly-default%3Acb1be24976e25734cb5fc13a8af6fdfb&seq=1#metadata_info_tab_contents

Abductive Thinking and Sensemaking: The Drivers of Design Synthesis by John Kolko, Design Issues, Vol. 26, No. 1 (Winter, 2010), pp. 15-28 (14 pages), Published by: The MIT Press

https://puniversity.informaticsglobal.com:2054/stable/20627839?Search=yes&resultItemClick=true&searchText=design+thinking&searchUri=%2Faction%2FdoBasicSearch%3FQuery%3Ddesign%2Bthinking%26so%3Drel&ab_segments=0%2FSYC-

6168%2Ftest&refreqid=fastly-default%3A0b89336ea274d63c010536b01316d7bb&seq=1#metadata_info_tab_contents

Designerly Ways of Knowing: Design Discipline versus Design Science by Nigel Cross, Design Issues, Vol. 17, No. 3 (Summer, 2001), pp. 49-55 (7 pages), Published by: The MIT Press

https://puniversity.informaticsglobal.com:2054/stable/1511801?Search=yes&resultItemClick=true&searchText=design+thinking&searchUri=%2Faction%2FdoBasicSearch%3FQuery%3Ddesign%2Bthinking%26so%3Drel&ab_segments=0%2FSYC-6168%2Ftest&refreqid=fastly-default%3A0d5b607b163f60876ca973ed90e22b1c&seq=1#metadata_info_tab_contents

Course Code: MAT1003	Course Title: Applied Statistics Type of Course: School Core	L-T-P-C	1	0	2	2
Version No.	3.0					
Course Pre-requisites	None					
Anti-requisites	None					
Course 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.					
Course 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.					

Expected 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.			
Module 1	Descriptive Statistics	Assignment	Coding 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 .				
Module 2	Probability			6 classes
Introduction to Probability, Probability of an event, Addition Principle, Multiplication law, Conditional Probability, Total Probability and Baye's theorem with examples				
Module 3	Random Variables and Probability Distributions		Coding 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				
Module 4	Sampling Theory		Coding needed	15 classes
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.
Targeted 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. Tools used: R Software / MS-Excel
Text Book Ronald E Walpole, Raymond H Myers, Sharon L Myers, and Keying E Ye, Probability and Statistics for Engineers and Scientists, Pearson Education, 2016.
References James T. McClave, P. George Benson and Terry Sincich, Statistics for Business and Economics, 2018. David R. Anderson, Dennis J. Sweeney, Thomas A. Williams, Essentials of Modern Business Statistics with Microsoft Excel, 2020. David R. Anderson, Dennis J. Sweeney, Thomas A. Williams, Essentials of Statistics for Business and Economics, 2019. Douglas C. Montgomery and George C. Runger, Applied Statistics and Probability for Engineers, John Wiley and Sons, 2018. Richard A. Johnson, Miller and Freund's Probability and Statistics for Engineers, 2018. Kishor S Trivedi, Probability and Statistics with reliability, Queuing and Computer Science Applications, John Wiley & Sons, 2008.

Topics 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.

Course Code: CHE1018	Course Title: Environmental Science	L- T- P- C	1	0	2	0
	Type of Course: School Core- Theory and Lab	Contact Hours	1	0	2	3
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 Determination of total alkalinity of a water sample (knowledge) Estimation of water hardness by EDTA method and its removal (by zeolite/ ion exchange method) (Comprehensive) Estimation of copper from industrial effluents by colorimetric method (Comprehensive) Estimation of iron from industrial effluents by titrimetric method/potentiometric method (Comprehensive) Estimation of nickel from industrial effluents by titrimetric method (Comprehensive) Estimation of chloride in drinking water by titrimetric method (Comprehensive) Estimation of fluoride in ground water by colorimetric method (Comprehensive) Determination of calcium in aqueous solution (Comprehensive) Determination 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 Midterm exam Assignment (review of digital/ e-resource from PU link given in references section - mandatory to submit screenshot accessing the digital resource.) Lab evaluation/Assignment End Term Exam

Self-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

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

Krishnamurthy, 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.

Sinha 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=DO_AB_1_06082022_18126

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO_AB_1_06082022_8761

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO_AJ_1_02082022_3333

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO_AB_1_06082022_3063

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO_AB_1_06082022_20719

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO_AB_1_06082022_16824

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO_AB_1_06082022_3954

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO_AB_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=SP_RINGER_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=TE_XTBOOK_LIBRARY01_06082022_395&xIndex=4

https://www.ugc.gov.in/oldpdf/modelcurriculum/env.pdf

Course Code: CIV1008	Course Title: Basic Engineering Sciences Type of Course: Theory Only	L-T-P-C	2	0	0	2
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course 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.					
Course Objective	The objective of the course is skill development of student by using Participative Learning techniques.					
Course 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.			
Course Content:				
Module 1	Introduction to various fields in Civil Engineering	Assignment	Case studies on different Civil Engineering Projects	6 Sessions
Topics: Introduction to Civil Engineering: Definition, scope and branches of Civil Engineering, Role of Civil Engineer, Overview of Infrastructure.				
Module 2	Current Trends and Evolution in Civil Engineering	Assignment	Article Review	6 Sessions
Topics: Mechanization in Construction, Application of Digital Technologies in Planning, Design, execution, monitoring and maintenance of Construction. Overview of Smart Cities.				
Module 3	Power Production and Consumption Machinery	Assignment & Quiz	Data Collection	6 Sessions
Topics: Energy and its types, Engines and their applications, Pumps-Compressors and their applications.				
Module 4	Overview of Petroleum Engineering	Assignment & Quiz	Article Review	6 Sessions
Overview 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				
Module 5	Industry 4.0	Assignment & Quiz	Data Collection	6 Sessions
Topics: Conventional manufacturing process: Metal forming, metal removal and metal joining process. Modern Manufacturing process: 3D Printing / Additive Manufacturing.				

Targeted Application & Tools that can be used:

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

Project work/Assignment:

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

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

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

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

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

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

Text 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

References

K.P. Roy, S.K. Hajra Choudhury, Nirjhar Roy, “Elements of Mechanical Engineering”, Media Promoters and Publishers Pvt Ltd, Mumbai.

Nontechnical Guide to Petroleum Geology, Exploration, Drilling & Production by Norman J. Hyne, PennWell Books; 3rd Revised edition

Web-resources:

Basic Civil Engineering

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

Post-parametric Automation in Design and Construction

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

Smart Cities : Introducing Digital Innovation to Cities

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

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

Mechanical Engineering

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO106_REDO_1705

Additive Manufacturing: Opportunities, Challenges, Implications

https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=1134464&site=ehost-live

Society of Petroleum Engineers (SPE)

https://www.spe.org/en/

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

Rigzone: A resource for news and information about the oil and gas industry, including job postings and industry trends.

https://www.rigzone.com/

Topics relevant to the development of SKILLS:

Engines-Turbines and their applications.

Mechanization in Construction.

Digitization in Petroleum Industries

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 https://youtube.com/playlist?list=PLu0W_9lII9agS67Uits0UnJyrYiXhDS6q https://puniversity.informaticsglobal.com:2229/login.aspx
Topics relevant to the development of "Skill Development": Static Polymorphism Method overloading, constructors constructor overloading this keyword static keyword and Inner classes Inheritance and Polymorphism. for Skill Development through Experiential Learning techniques. This is attained through the assessment component mentioned in the course handout.

ENG2001	Advanced English	L- T- P- C	1	0	2	2
Version No.	1.3					
Course Pre-requisites	ENG1002 Technical English					
Anti-requisites	NIL					
Course 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.					

Course Out Come	On successful completion of the course the students shall be able to: Develop a critical and informed response reflectively, analytically, discursively, and creatively to their reading. Communicate effectively, creatively, accurately and appropriately in their writing. Deliver technical presentations Design resume and create professional portfolio to find a suitable career			
Course Content: Theory				
Module 1	Critical Reasoning and Writing	Writing Essays	Critical Reading	4 Classes
Topics: 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)				
Module 2	Technical Presentation	Presentation	Oral Skills	3 Classes
Topics: Planning the presentation Creating the presentation Giving the presentation				
Module 3	Writing Reviews	Prezi	Review Writing	4 Classes
Topics: Review Writing Short film reviews Advanced English Grammar (Self Study)				

Module 4	Starting your Career	Online Writing Lab	Writing Skills	4 Classes
Topics: Preparing a Resume Writing Effective Application Letter Creating a Professional Portfolio				
Course Content: Practical Sessions				
Module 1	Critical Reasoning and Writing			8 Classes
Reading and Analyzing Level 1 – Annotation Level 2 - Assumptions Writing Narrative Essays Level 1 – Draft 1 Level 2 – Draft 2				
Module 2	Technical Presentation			10 Classes
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 Technical Group Presentation				
Module 3	Writing Reviews			Classes
Practice Worksheets Level 1 – Eliminating the Passive Voice Level 2 – Simple, compound and complex sentences				

Writing Short Film Reviews		
Module 4	Starting your Career	Classes
Collaborative Project Job search and writing report Writing Resume		
Module 1-4	Academic Journal	2 Classes
Academic Journal Writing Level 1- Mid Term Level 2 – End Term		
Targeted Application & Tools that can be used: Writing reports, Review writing, Group Discussion, Dyadic interviews, Grammarly.com		
Project 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.		
References		

Hering, Heik. How to Write Technical Reports: Understanding Structure, Good Design, Convincing Presentation. Springer.

Johnson, Richard. (2010) Technical Communication Today. Pearson, 2015

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.

The Princeton Review. (2010) MCAT Verbal Reasoning & Writing. The Princeton Review, Inc.

https://www.hitbullseye.com/Strong-and-Weak-Arguments.php Accessed on 10 Dec 2021

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

Course Code: PPS1012	Course Title: Enhancing Personality through Soft Skills Type of Course: Practical Only Course	L- T - P- C	0	0	2	1
Version No.	1.0					
Course Pre-requisites	Students are expected to understand Basic English. Students should have desire and enthusiasm to involve, participate and learn.					
Anti-requisites	NIL					
Course 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.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of “Personality Development through Soft Skills” and attain SKILL DEVELOPMENT through PARTICIPATIVE LEARNING techniques.					

<table>
<tr><td></td><td colspan="3"></td></tr>
<tr><td>Course Out Comes</td><td colspan="3">On successful completion of this course the students shall be able to:
CO 1 Identify the stages of team formation (Remember)
CO 2 Demonstrate effective presentation skills (Apply)
CO3 Prepare professional social media profile (Apply)</td></tr>
<tr><td>Course Content:</td><td colspan="3"></td></tr>
<tr><td>Module 1</td><td>Team Building</td><td>Classroom and outbound team building activities.</td><td>6 Hours</td></tr>
<tr><td colspan="4">Topics: Importance of team, stages of Team Formation, Trust and collaboration, Virtual Team.
Activity: Team Building outbound activity</td></tr>
<tr><td>Module 2</td><td>Art of Questioning</td><td>Role plays</td><td>4 Sessions</td></tr>
<tr><td colspan="4">Topics: Framing Questions, 5W1H Technique, Open-ended and Close-ended questions, Funnel technique, Probing questions, Leading questions</td></tr>
<tr><td>Module 3</td><td>Presentation Skills</td><td>Practice and evaluation of individual / group presentation</td><td>10 Sessions</td></tr>
<tr><td colspan="4">Topics: Content development, Delivery techniques, Audience Analysis, Timing and Pacing, handling questions and challenges.
Activity: Individual presentations and team presentation</td></tr>
</table>

Module 4	Professional Brand Building	Brand Framework Activity	4 Sessions
Topics: Personal brand definition, Crafting a compelling LinkedIn profile, Networking strategies. Activity: Create a basic online profile			
Module 5	Recap / Revision /Feedback Session		1 Session
Targeted Application & Tools that can be used: TED Talks You Tube Links Activities			
Project work/Assignment: Mention the Type of Project /Assignment proposed for this course			
Presentation Evaluation			
Targeted Application & Tools that can be used: TED Talks YouTube Links Videos by L&D Team shared on Edhitch/YouTube.com LMS Assignments proposed for this course Evaluation on Presentation Assignment on LinkedIn Post			

YouTube Links: https://youtu.be/z__jxoczNWc (Steve Jobs Introducing the iPhone 4 in June 2010)

References

"Talk Like TED - The 9 Public-Speaking Secrets of the World's Top Minds" By Carmine Gallo St. Martin's Press Copyright © 2014 Carmine Gallo All rights reserved. ISBN: 978-1-250-04112-8

"The Presentation Secrets of Steve Jobs: How to Be Insanely Great in Front of Any Audience" MP3 CD – Import, 22 April 2014

"The Definitive Book of Body Language: The Hidden Meaning Behind People's Gestures and Expressions" Hardcover – Illustrated, 25 July 2006

"Crucial Conversations: Tools for Talking When Stakes Are High" Paperback – Import, 1 July 2002

Web links:

https://www.wordstream.com/blog/ws/2014/11/19/how-to-improve-presentation-skills

https://www.cbs.de/en/blog/15-effective-presentation-tips-to-improve-presentation-skills/

https://hbr.org/2022/05/the-art-of-asking-great-questions

Topics relevant to development of "SKILL": Art of Presentation, Team building, Art of questioning and Personal Branding for Skill Development through Participative Learning Techniques. This is attained through assessment component mentioned in course handout.

Course Code: EEE1007	Course Title: Basics of Electrical and Electronics Engineering. Type of Course: Engineering Science - Theory & Integrated Laboratory	L-T-P-C	3	0	2	4
Version No.	1.0					

Course Pre-requisites	NIL			
Anti-requisites	NIL			
Course Description	This is a fundamental Course which is designed to know the use of basics of electrical and electronics engineering principles occurs in various fields of Engineering. The course emphasis on the characteristics and applications of Electrical and Electronics devices, working, analysis and design of electrical circuits using both active & passive components, fundamentals of electrical machines and basics of transistors and its application. The associated laboratory provides an opportunity to validate the concepts taught and enhances the ability to visualize the real system performance, using both hardware and simulation tools.			
Course Objective	The objective of the course is to familiarize the learners with the concepts of Basics of Electrical and Electronics Engineering and attain Skill Development through Experiential Learning techniques.			
Course Outcomes	On successful completion of this course the students shall be able to: Explain basic laws of Electrical Engineering to compute voltage, currents and other parameters in the circuits. Discuss various fundamental parameters appearing in the characteristics of semiconductor devices and their applications. Summarize the operations of different biasing configurations of BJTs and amplifiers. Summarizethe performance characteristics and applications of various electrical Machines. Demonstrate the working of electrical machines to observe performance characteristics Demonstrate the working of electronic circuits to obtain the V-I Characteristics of various semiconductor devices.			
Course Content:				
Module 1	Introduction to Electrical Circuits	Assignment/ Quiz	Numerical solving Task	10 Sessions

DC Circuits: Concept of Circuit and Network, Types of elements, Network Reduction Techniques- Series and parallel connections of resistive networks, Star–to-Delta Transformations, Mesh Analysis, Numerical examples. AC Circuits: Fundamentals of single phase circuits - Series RL, RC and R-L-C Circuits, Concept of active power, reactive power and Power factor, Numerical examples. Introduction to three phase system and relation between line and phase values in Star & Delta connection, Numerical examples.				
Module 2	Semiconductor and Diode applications	Assignment/ Quiz	Memory Recall based Quizzes	11 Sessions
Mass Action Law, Charge densities in a semiconductor, Types of SC, Junction diodes -Ideal and practical behaviour, Modelling the Diode Characteristic, and Diode applications like rectifiers, Clipping and clamping circuits. Zener diode, characteristics and its applications like voltage regulator.				
Module 3	Fundamentals of Electrical Machines	Assignment/ Quiz	Memory Recall-based Quizzes	12 Sessions
Electrical Machines: Single phase transformers: principle of operation and EMF equation, Numerical examples. DC Motor: principle of operation, Back EMF, torque equation, Numerical examples. AC Motor: Principle operation of Induction Motors and its Applications. Special Machines: Introduction to special electrical machines and its applications.				
Module 4	Transistors and its Applications	Assignment/ Quiz	Numerical solving Task	12 Sessions
Transistor characteristics, Current components, BJT Configurations (CB, CC, CE configurations) and their current gains. Operating point, Biasing & stabilization techniques: Fixed Bias, Voltage divider bias and its stability factor and load line analysis. Single and multistage amplifier, Darlington pair. JFET (Construction, principal of Operation and Volt –Ampere characteristics). Pinch- off voltage, Comparison of BJT and FET. MOSFET (Construction, principal of Operation and symbol), MOSFET characteristics in Enhancement and Depletion modes.				
List of Laboratory Tasks: Experiment No 1: Verification of KVL and KCL for a given DC circuit.				

Level 1: Study and Verify KVL and KCL for the given electrical Circuit.

Level 2: For the same circuit considered in level 1, perform the simulation using NI LabVIEW/Multisim/MATLAB.

Experiment No 2: Analyse AC series circuits – RL, RC and RLC .

Level 1: Conduct an experiment to perform and verify the impedance, current and power of Series RL and RC circuits

Level 2:

Experiment No 3: Calculation of power and power factor of the given AC Circuit.

Level 1: Conduct an experiment to measure the power and power factor for given resistive load.

Level 2: Conduct an experiment to measure the power and power factor for given inductive load.

Experiment No 4: Perform the experiments on given Transformer.

Level 1: Verify the EMF equation of a transformer and compute the voltage transformation ratio.

Level 2: Study the effect of load on the secondary side of the transformer and verify the EMF equation under load conditions.

Experiment 5: Load test on DC shunt motor

Level 1: Conduct load test on DC shunt motor and find its efficiency at different loads

Level 2:Conduct load test on DC shunt motor and plot the performance characteristics.

Experiment 6: Study of PN-Junction Diode Characteristics in Forward and Reverse Bias Conditions.

Level 1:Carry out an experiment to plot VI Characteristics and hence find the cut-in voltage on forward characteristics for the Silicon P-N Junction diode.

Level 2: Carry out an experiment to plot VI Characteristics of Zener diode and hence find the zener voltage on reverse characteristics for the Silicon P-N Junction zener diode.

Experiment 7: To observe the output waveform of half wave and full wave rectifier circuit and compute ripple factor and efficiency

Level 1:Identify the components required for a rectifier circuit, rig up the circuit, and sketch the output waveforms without filter.

Level 2: Rig up the rectifier circuit with RC filter, observe the output waveforms, determine the efficiency and ripple factor.

Experiment 8: To construct clipping and clamping circuits for different reference voltages and to verify the responses.

Level 1:Identify the components required for building a Clipper / Clamper circuit. Rig up the circuit according to the circuit diagram given and sketch the output waveform.

Level 2: Given a sinusoidal input of 10 V p-p, implement a positive / negative clipper with output clipped at 2 V.

Experiment 9: To calculate various parameters of emitter follower circuit using BJT

Level 1: Identify the components required to implement an emitter follower circuit. Rig up the circuit and observe the variations in output waveform with respect to the variations in input waveform.

Level 2: Determine the values of Zin input impedance and Zout output impedance for Emitter Follower.

Experiment 10: To Implement RC Coupled amplifier using a BJT and sketch the frequency response.

Level 1: Identify the components required to implement an RC coupled amplifier circuit. Rig up the circuit and sketch the frequency response.

Level 2: From the frequency response curve determine the value of the mid band gain and the bandwidth.

Targeted Application & Tools that can be used:

Targeted Applications: Application Area includes all electrical and electronic circuits (power supply unit, regulator unit, embedded devices, hardware electronics etc.). The students will be able to join a profession which involves basics to high level of electronic circuit design.

Professionally Used Software: Matlab/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..

Text Book(s):

Kothari D. P. &Nagrath I. J., "Basic Electrical and Electronics Engineering", Tata McGraw-HillEducation.

Theraja B.L. and Theraja A.K., "A Textbook of Electrical Technology: Basic Electrical Engineering" in S.I. System of Units, 23rd ed., New Delhi: S. Chand, 2002.

A.P.Malvino, Electronic Principles,7thEdition, Tata McGraw Hill,2007

J. Millman, C. C. Halkias and C. D. Parikh, "Millman's Integrated Electronics", McGraw Hill Education, 2nd Edition.

Basics of Electrical & Electronics Laboratory Manual.

Reference Book (s):

John Hiley, Keith Brown and Ian McKenzie Smith, "HUGHES Electrical and Electronic Technology", 10th Edition (Indian Edition published by Dorling Kindersley), Pearson,2011

Samarajit Ghosh, "Fundamentals of Electrical and Electronics Engineering", 2nd Edition, Prentice Hall India, 2007.

K Uma Rao, A Jaya Lakshmi, "Basic Electrical engineering" IK International publishing house Pvt. Ltd

R. L. Boylestad and L. Nashelsky, "Electronic Devices and Circuit Theory", Pearson Education India 7th Edition.

A K. Maini, V. Agrawal, "Electronic Devices & Circuits", Wiley, 2nd Edition

A.S Sedra, K. C. Smith, "Microelectronic Circuits", Oxford University Press, 6th Edition

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

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

https://www.digimat.in/nptel/courses/video/108105112/L01 "Fundamentals of Electrical Engineering-Basic Concepts, Examples"

Seminar Topic: https://nptel.ac.in/courses/108/105/108105153/ “Electrical Measurements”

Video lectures on “Electronic Devices” by Prof.Dr. A. N. Chandorkar, IIT Bombay http://www.satishkashyap.com/2013/03/video-lectures-on-electron-devices-by.html

Video lectures on “Analog Electronics” by Prof. S.C. Dutta Roy, IIT Delhi https://nptel.ac.in/courses/108/102/108102095/

Video lectures on “Diodes”, by Prof.ChitralekhaMahanta, IIT Guwahati,

https://nptel.ac.in/courses/117/103/117103063/

“Introduction to Electrical Machines https://nptel.ac.in/courses/108/102/108102146/”

M. -Y. Kao, H. Kam and C. Hu, "Deep-Learning-Assisted Physics-Driven MOSFET CurrentVoltageModeling," in IEEE Electron Device Letters, vol. 43, no. 6, pp. 974-977, June 2022, doi: 10.1109/LED.2022.3168243

https://ieeexplore-ieee-org-resiuniv.knimbus.com/document/9758727

F. Bonet, O. Aviñó-Salvadó, M. Vellvehi, X. Jordà, P. Godignon and X. Perpiñà, "Carrier

Concentration Analysis in 1.2 kV SiCSchottky Diodes Under Current Crowding," in IEEE Electron DeviceLetters, vol. 43, no. 6, pp. 938-941, June 2022, doi: 10.1109/LED.2022.3171112. https://ieeexplore-ieeeorg-presiuniv.knimbus.com/document/9764749

M. Chanda, S. Jain, S. De and C. K. Sarkar, "Implementation of Subthreshold Adiabatic Logic for Ultralow-Power Application," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. .23, no. 12, pp. 2782-2790, Dec. 2015.

https://ieeexplore.ieee.org/document/7018053

R. Raut and O. Ghasemi, "A power efficient wide band trans-impedance amplifier in submicron

CMOS integrated circuit technology," 2008 Joint 6th International IEEE Northeast Workshop on Circuits and Systems and TAISA Conference, 2008, pp. 113-116, doi: 0.1109/NEWCAS.2008.4606334. https://ieeexplore.ieee.org/document/4606334

Topics relevant to “SKILL DEVELOPMENT”: Performing suitable experiments to compute the Electrical and electronics circuit parameters, performance operation of Machines, and semiconductor devices for Skill Developmentthrough Experiential Learning techniques. This is attained through assessment component mentioned in course plan.

Course Code: CSE1006	Course Title: Problem Solving using JAVA Type of Course: 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 on 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 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 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.				

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 https://youtube.com/playlist?list=PLu0W_9lII9agS67Uits0UnJyrYiXhDS6q https://puniversity.informaticsglobal.com:2229/login.aspx
Topics relevant to development of "Skill Development": Static Polymorphism Method overloading, constructors constructor overloading this keyword static keyword and Inner classes Inheritance and Polymorphism. for Skill Development through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: FRL1002	Course Title: Basic French Type of Course: Open Elective	L- T-P- C	2-0-0-2
Version No.	4.0		
Course Pre-requisites	Not Applicable		
Anti-requisites	Not Applicable		

Course Description	This Course is for beginners and gives an introduction of the French Language (basic grammar, conjugation, daily used vocabulary words, and basic conversations) and French culture. This Course is designed to build up all of the basic skills of French listening, reading, speaking, and writing introduced in the lessons. Besides, this Course offers an access to the French world, helping students to break cultural boundaries and raise cultural literacy.			
Course Objective	This course is designed to improve the learners Employability skills by using participative learning techniques to develop students' language proficiency and cross-cultural competence by active and participatory teaching methods.			
Course Outcomes	On successful completion of the course the students shall be able to: Identify the basics of French Grammar, vocabulary and Conjugation Apply the basics strategies of listening, reading, speaking and writing skills Use of French on everyday topics such as greetings, personal information, time and schedule Practice conversations in French language with peer speakers in different situations			
Course Content:	Learning of Basic French skills			
Module 1	Greetings and Introducing yourself		[Remember]	6 Periods
Chapter 1. Greetings Objectives: Greetings, introducing yourself, how to welcome someone, Grammar: Construction of a sentence, the days of the weeks and the months Chapter 2. Introducing yourself Objectives: Introduce oneself / ask for someone's personal information. Grammar: Mas or Fem noun, adjectives, present tense of the 1st group Usage of audio visual files				

Module 2	Expressing likes/dislikes and introducing someone		[Apply]	6 Periods
Chapter 3. Expressing likes and dislikes Objectives: How to expressing what you like and dislike. Grammar: Negative form, singular and plural. Culture: The polite way to address people in French Assignment Chapter 4. Introducing someone Objectives: How to describe someone, Grammar: Vocabulary of the family, Demonstrative adjectives, Present tense of verbs of the 2nd and 3rd group				
Module 3	Inviting someone and asking questions		[Apply]	9 Periods
Chapter 5. Inviting someone Objectives: How to invite someone, accept or refuse the invitation, Read the time, Grammar: Future tense, Interrogation. Culture: The art of accepting and declining an invitation politely in French Internal Chapter 6. Asking for information, Objectives: How to ask for information, giving information				
Module 4	Making a reservation and giving directions		[Apply]	9 Periods
Chapter 7: Making a Reservation Objectives: How to make a reservation, future tense Chapter 8 : Giving directions Objectives: How to ask for directions, Imperative tense				

Group discussions
Targeted Application & Tools that can be used
Project work /Assignments Assignment (Essay writing / presentation) Internal Group work / Group discussions
Text Book L'Atelier 1 - - Méthode de Français--- Niveau A1 (Didier – 2019) Festival 1- - Méthode de Français--- Niveau A1 (CLE International – 2005)
References Learning materials designed by the instructor
Topics relevant to development of 'Employability Skills' through participative learning techniques: Foreign language proficiency and cross-cultural competence by active and participatory teaching methods.

Course Code: ECE2010	Course Title: Innovative Projects using Arduino		L- T-P- C	-	-	-	1
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course 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.					
Course Objective	The objective of the course is Employability Skills of student by using PARTICIPATIVE LEARNING techniques.					
Course Outcomes	On successful completion of the course the students shall be able to Explain the main features of the Arduino prototype board Demonstrate the hardware interfacing of the peripherals to Arduino system. Understand the types of sensors and its functions Demonstrate the functioning of live projects carried out using Arduino system.					
Course Content:						
Module 1	Basic concepts of Arduino	Hands-on	Interfacing Task and Analysis	4 Sessions		
Topics:						

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.				
Module 2	Sensory Devices	Hands-on	Interfacing Task and Analysis	4 Sessions
Arduino Sensors: Humidity Sensor, Temperature Sensor, Water Detector / Sensor, PIR Sensor, Ultrasonic Sensor, Connecting Switches and actuators, sensor interface with Arduino. Introduction 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				
Targeted Application & Tools that can be used: Application Area: Home 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. Professionally Used Software: students can use open SOURCE Softwares Arduino IDE and Tincker CAD				
Project 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 .

3. 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

Textbook(s):

Monk Simon “Programming Arduino: Getting Started with Sketches”, Mc Graw Hill Publications Second Edition

References

Reference 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.

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

Arduino trending Projects < https://www. https://projecthub.arduino.cc/>

Introduction to Arduino < https://onlinecourses.swayam2.ac.in/aic20_sp04/preview>

Case studies on Wearable technology< https://www.hticiitm.org/wearables>

E-content:

Cattle Health Monitoring System Using Arduino and IOT (April 2021| IJIRT | Volume 7 Issue 11 | ISSN: 2349-6002)

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.

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.

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.

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

Course Code: MAT2501	Course Title: Integral Transforms and Partial Differential Equations Type of Course:1] School Core	L-T- P- C	3	0	0	3
Version No.	1.0					
Course Pre-requisites	Calculus and Differential Equations					
Anti-requisites	NIL					
Course 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 solutions of different 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.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of "Transform Techniques, Partial Differential Equations" and attain Skill Development through Problem Solving Techniques.					
Course Out Comes	On successful completion of the course the students shall be able to: CO1 - Express functions in terms of uniformly convergent Fourier series. CO2 - Apply Laplace transform technique to solve differential equations. CO3 - Employ Z-transform techniques to solve difference equations. CO4 - Solve a variety of partial differential equations analytically.					
Course Content:						
Module 1	Laplace Transforms		(12 Classes)			
Definition and Laplace transform of elementary functions. Properties of Laplace transform, and Laplace transform of periodic function, unit-step function and Impulse function – related problems. Inverse Laplace transform of standard functions - problems, initial and final value						

theorem. Convolution theorem, solution of linear and simultaneous differential equations and LCR Circuit.			
Module 2	Fourier Series	Assignment	(8 Classes)
Fourier Series: Periodic functions, Dirichlet's condition. Fourier series of periodic functions period 2π and arbitrary period. Half range Fourier series. Practical harmonic analysis.			
Module 3	Fourier Transforms and Z - Transforms		(13 Classes)
Fourier Transforms: Definitions, infinite Fourier transforms, Fourier sine and cosine transforms, inverse Fourier transforms, Problems. Difference equations and Z-transforms: Z-transforms – Basic definitions, Standard Z-transforms, Linearity property, Damping rule, Shifting rule, Initial value theorem, Final value theorem, Inverse Z-transforms. Difference equations – Basic definitions, Application of Z-transforms to solve difference equations.			
Module 4	Partial Differential Equations	Assignment	(12 Classes)
Formation of PDE, Solution of non-homogeneous PDE by direct integration, Solution of homogeneous PDE involving derivative with respect to one independent variable only (Both types with given set of conditions) Method of separation of variables. (First and second order equations) Solution of Lagrange's linear PDE. of the type $P\,p + Q\,q = R$. Applications of PDE: Derivation of one-dimensional wave and heat equations. Various possible solutions of these by the method of separation of variables. D'Alembert's solution of wave equation. Two-dimensional Laplace's equation – various possible solutions. Solution of all these equations with specified boundary conditions (Boundary value problems).			
Targeted 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 to equip them with the necessary numerical approaches and basic statistical tools to tackle engineering and real-life problems.			
Assignment:			
Newton-Raphson Methods, Gauss-Seidel Method, LU Decomposition, Trapezoidal Rule, Simpson's rule, Runge-Kutta 4th Order.			
Text Book Erwin Kreyzig, Advanced Engineering Mathematics, John Wiley and sons, Inc.10th Edition			

B. S. Grewal (2017), Higher Engineering Mathematics by, 44th Edition, Khanna Publishers.

References:

Victor Henner, Tatyana Belozerova, Mickhail Khenner, Ordinary and Partial Differential Equations, CRC Press, Edition, 2013.

Walter Ledermann, Multiple integrals, Springer, 1st edition

E-resources/ Web links:

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_140238

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_233298

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_204892

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_246791

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_223548

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_134719

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_32614

https://www.math.hkust.edu.hk/~maqian/ma006_0607F.html

https://www.scu.edu.au/study-at-scu/units/math1005/2022/

Topics relevant to SKILL DEVELOPMENT: The course focuses on the concepts of calculus and differential equation with reference to specific engineering problems. The course is of both conceptual and analytical type in nature through Problem solving. This is attained through the assessment component mentioned in course handout.

Course Code: CBC1700

Course Title: Foundations of Blockchain Technology
L:T:P:C – 3:0:0:3

Course Description
This course introduces the foundational principles of blockchain technology, its architecture, components, and applications. It explores the working of distributed ledgers, cryptographic fundamentals, consensus algorithms, and the structure of cryptocurrencies and smart contracts.

Course Objectives
— Understand the fundamentals of distributed ledger technology
— Explore cryptographic primitives used in blockchain systems
— Explain consensus algorithms and their roles in decentralized networks
— Analyze the architecture and components of blockchain platforms
— Examine blockchain applications across industries

Course Outcomes
CO1 (Understand): Describe the architecture and components of blockchain technology
CO2 (Understand): Explain cryptographic techniques and consensus mechanisms in blockchain
CO3 (Apply): Demonstrate the creation and validation of transactions and blocks
CO4 (Analyze): Compare various blockchain platforms and their real-world applications

---

Course Content (45 Hours Total)

Module 1: Introduction to Blockchain Technology – 10 Sessions
History of blockchain, Evolution from Bitcoin to Web3, Distributed ledger technology, Key characteristics: immutability, transparency, trust, Use cases and applications

---

Module 2: Cryptography and Blockchain – 12 Sessions
Hash functions (SHA-256), Digital signatures, Merkle trees, Public and private key cryptography, Wallets and addresses, Transaction lifecycle

---

Module 3: Consensus Mechanisms – 11 Sessions
Consensus overview, Proof-of-Work (PoW), Proof-of-Stake (PoS), Practical Byzantine Fault Tolerance (PBFT), Delegated Proof of Stake (DPoS), Comparison of consensus algorithms

---

Module 4: Blockchain Platforms and Applications – 12 Sessions
Bitcoin overview, Ethereum overview, Permissioned vs permissionless blockchains,

Hyperledger Fabric basics, Smart contracts, Blockchain in finance, healthcare, and supply chain

---

Textbooks

T1: Narayanan et al., *Bitcoin and Cryptocurrency Technologies*, Princeton University Press

T2: Imran Bashir, *Mastering Blockchain*, Packt Publishing, 3rd Edition, 2020

Reference Books

R1: Antonopoulos, *Mastering Bitcoin*, O'Reilly Media

R2: Arvind Narayanan et al., *Introduction to Cryptography and Blockchain*, Princeton Press

R3: Don Tapscott & Alex Tapscott, *Blockchain Revolution*, Portfolio

R4: Josh Thompson, *Blockchain Basics*, CreateSpace

Web Resources

W1: https://blockgeeks.com

W2: https://ethereum.org

W3: https://bitcoin.org

W4: https://hyperledger.org

W5: https://web3.foundation

Course Code: CBD1700
Course Title: Introduction to Big Data
L:T:P:C – 3:0:0:3

Course Description
This course introduces the fundamentals of big data, covering its characteristics, architecture, and the ecosystem of tools used for storage, processing, and analytics. It emphasizes the challenges and techniques for handling large-scale structured and unstructured data.

Course Objectives
— Understand the key concepts and characteristics of big data
— Analyze big data frameworks, tools, and ecosystems
— Apply basic data processing operations using Hadoop and MapReduce
— Explore storage, retrieval, and processing strategies for massive datasets

---

Course Outcomes
CO1 (Understand): Describe the characteristics and challenges of big data
CO2 (Analyze): Compare big data technologies and processing frameworks
CO3 (Apply): Execute basic data operations using Hadoop and MapReduce
CO4 (Apply): Analyze use cases and architectures for big data applications

---

Course Content (45 Hours Total)

Module 1: Big Data Fundamentals – 11 Sessions
Definition and evolution of Big Data, Characteristics: Volume, Velocity, Variety, Veracity, and Value, Big Data vs Traditional Data, Applications and trends

---

Module 2: Big Data Architecture and Storage – 11 Sessions
Big Data architecture components, HDFS: concepts and architecture, File formats (CSV, JSON, Avro, Parquet), Data ingestion tools (Sqoop, Flume)

Module 3: Big Data Processing – 11 Sessions
MapReduce programming model, Hadoop ecosystem, Data flow, YARN architecture, Job scheduling and optimization

---

Module 4: Big Data Ecosystem and Analytics – 12 Sessions
Apache Spark overview, RDDs and DataFrames, Hive and Pig basics, Big Data Analytics use cases, Real-time streaming with Kafka

---

Textbooks

T1: Seema Acharya, Subhasini Chellappan, *Big Data and Analytics*, Wiley India

T2: Tom White, *Hadoop: The Definitive Guide*, O'Reilly Media, 4th Edition

---

Reference Books

R1: Vignesh Prajapati, *Big Data Analytics with R and Hadoop*, Packt Publishing

R2: Alex Holmes, *Hadoop in Practice*, Manning Publications

R3: Chuck Lam, *Hadoop in Action*, Manning Publications

R4: Alan Gates, *Programming Pig*, O'Reilly Media

Web Resources

W1: https://hadoop.apache.org

W2: https://spark.apache.org

W3: https://kafka.apache.org

W4: https://data-flair.training

W5: https://www.tutorialspoint.com/big_data_analytics

Course Code: CSE1508	**Course Title:** Data Structures **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 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 SKILL DEVELOPMENT of student by using EXPERIENTIAL LEARNING techniques					

<table>
<tr><td>Course Out Comes</td><td colspan="4">On successful completion of the course the students shall be able to:

CO1 :Describe the concept of basic data structure, stacks, queues, and arrays and their operations. [Understand]

CO2: Utilize linked lists for real-time scenarios. [Apply]

CO3: Apply an appropriate non-linear data structure for a given scenario. [Apply]

CO4: Demonstrate different searching and sorting techniques. [Apply]</td></tr>
<tr><td>Course Content:</td><td colspan="4"></td></tr>
<tr><td>Module 1</td><td>Introduction to Data Structure and Linear Data Structure –Stacks and Queues</td><td>Assignment</td><td>Program activity</td><td>9 Hours</td></tr>
<tr><td colspan="5">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.</td></tr>
<tr><td>Module 2</td><td>Linear Data Structure -Linked List</td><td>Assignment</td><td>Program activity</td><td>12 Hours</td></tr>
<tr><td colspan="5">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.</td></tr>
<tr><td>Module 3</td><td>Non-linear Data Structures - Trees</td><td>Assignment</td><td>Program activity</td><td>12 Hours</td></tr>
<tr><td colspan="5">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.Red Black Tree, Expression Tree , Heaps -AVL Trees ,Binary Serach Tree ,</td></tr>
<tr><td>Module 4</td><td>Non-linear Data Structures -</td><td>Assignment</td><td>Program activity</td><td>6 Hours</td></tr>
</table>

	Graphs and Hashing			
Topics: Graphs: Basic Concept of Graph Theory and its Properties, Representation of Graphs . ADT, Elementary graph operations, Minimum Cost spanning trees, Shortest path and Transitive closure. **Hashing**: Introduction, Static Hashing, Dynamic Hashing				
Module 5	**Searching & Sorting**	Assignment	Program activity	**6 Hours**
Topic: Sorting & Searching - Sequential and Binary Search, Sorting – Selection and Insertion sort, Quick sort, Merge Sort, Bubble sort.				
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 on Stack application – Evaluation of postfix **Lab sheet -5** **Level 1:** Programming Exercises on Queues and its operations with conditions **Level 2:** - **Lab sheet -6** **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 -7

Level 1: Programming Exercises on Circular Linked list and its operations.

Level 2: Programming Exercises on Circular Linked list and its operations with various positions

Lab sheet -8

Level 1: Programming Exercises on factorial of a number

Level 2: Programming the tower of Hanoi using recursion

Lab sheet -9

Level 1: -

Level 2: Programming the tower of Hanoi using recursion

Lab sheet -10

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

Level 2: -

Lab sheet -11

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 -12

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

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

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 Insertion Sort

Lab sheet -14 (Beyond syllabus activity)
Level 1: Program to Construct AVL Tree
Level 2:
Lab sheet -15 (Beyond syllabus activity)
Level 1: Program to Construct RED BLACK Tree

Targeted Application & Tools that can be used

Use of PowerPoint software for lecture slides and use of Modern IDE like VS Code and Eclipse for lab programs to execute.

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 Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Edition, Universities Press, reprint 2018.

T2 Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.

References

R1 Data structures and program design in C by Robert Kruse, Tondo C L, Bruce Leung, Pearson education publishers, 2017.

R2 Programming and Data Structure by Jackulin C Salini etal., Ane books publishers, 2019.

Web resources:

1. For theory :https://onlinecourses.nptel.ac.in/noc20_cs85/preview
2. https://puniversity.informaticsglobal.com/login

Topics relevant to development of "Skill Development":

Linked list and stacks

Topics relevant to development of "Environment and sustainability: Queues

Course Code: MAT2605	**Course Title: Discrete Mathematics** **Type of Course:1] School Core**	**L-T- P-C**	**4**	**0**	**0**	**4**
Version No.	1.0					

Course Pre-requisites		Linear Algebra		
Anti-requisites		**NIL**		
Course Description		The course explores the study of mathematical structures that are fundamentally discrete (not continuous), focusing on concepts like set theory, logic, graph theory, combinatorics, and number theory, with applications primarily in computer science fields like algorithms, software development, and cryptography; it covers topics such as propositional logic, proof techniques, relations, functions, counting principles, and basic graph algorithms, providing a foundation for analyzing discrete problems and structures within computer science.		
Course Objective		The main objective of the course is that students should learn a particular set of mathematical facts and how to apply them. It teaches students how to think logically and mathematically through five important themes: mathematical reasoning, combinatorial analysis, discrete structures, algorithmic thinking, and applications and modeling. A successful discrete mathematics course should carefully blend and balance all five themes.		
Course Outcomes		On successful completion of the course the students shall be able to: CO1 - Explain logical sentences through predicates, quantifiers and logical connectives. CO2 - Deploy the counting techniques to tackle combinatorial problems CO3 - Comprehend the basic principles of set theory and different types of relations. CO4 - Apply different types of structures of trees for developing programming skills		
Course Content:				
Module 1	**Fundamentals of Logic**			**(10 Classes)**
Basic Connectives and Truth Tables, Propositional Logic, Applications of Propositional Logic, Propositional Equivalences, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference, Introduction to Proofs, Proof Methods and Strategy.				
Module 2	**Principle of Counting**	**Assignment**		**(15 Classes)**

The Well Ordering Principle – Mathematical Induction

The Basics of Counting, Permutations and Combinations, Binomial Coefficients and Identities, Generalized Permutations and Combinations, Generating Permutations and Combinations

Advanced Principle Counting: The Principle of Inclusion and Exclusion, Generalizations of the Principle, Derangements – Nothing is in its Right Place, Rook Polynomials.

Module 3	**Relations and Functions**		**(10 Classes)**

Cartesian Products and Relations, Functions, One-to-One, Onto Functions. The Pigeon-hole Principle, Function Composition and Inverse Functions.

Relations, Properties of Relations, Computer Recognition – Zero-One Matrices and Directed Graphs, Partial Orders, Lattice, Hasse Diagrams, Equivalence Relations and Partitions.

Module 4	**Recurrence Relations and Generating Functions**		**(10 Classes)**

Homogeneous and inhomogeneous recurrences and their solutions - solving recurrences using generating functions - Repertoire method - Perturbation method - Convolutions - simple manipulations and tricks.

Module 5	**Graph Theory & Algorithms on Networks**	**Assignment**	**(15 Classes)**

Definitions and basic results - Representation of a graph by a matrix and adjacency list - Trees - Cycles - Properties - Paths and connectedness - Sub graphs - Graph Isomorphism - Operations on graphs - Vertex and edge cuts - Vertex and edge connectivity, Euler and Hamilton Paths, Shortest-Paths.

Tree - Definitions, Properties, and Examples, Routed Trees, Binary search tree, Decision tree, spanning tree: BFS, DFS.

Algorithms on Networks - Shortest path algorithm- Dijikstra's algorithm, Minimal spanning tree- Kruskal algorithm and Prim's algorithm.

Targeted Application & Tools that can be used:

Discrete 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.

Assignment:

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

Text Book

1. Kenneth H. Rosen, “Discrete Mathematics and its Applications”, McGraw-Hill,s 8th Edition,2019.
2. Harary – Graph Theory, Addison-Wesley Publishing Company.

References:

1. Arthur Gill, "Applied Algebra for Computer Science", Prentice Hall.
2. K.D. Joshi, "Discrete Mathematics", Wiley Eastern Ltd.
3. Ralph. P. Grimaldi., “Discrete and Combinatorial Mathematics: An Applied Introduction”, 4th Edition, Pearson Education Asia.

E-resources/ Web links:

1. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_54588
2. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_375
3. https://www.math.hkust.edu.hk/~maqian/ma006_0607F.html
4. https://www.scu.edu.au/study-at-scu/units/math1005/2022/
5.

Topics relevant to SKILL DEVELOPMENT: The course focuses on the concepts of calculus and differential equation with reference to specific engineering problems. The course is of both conceptual and analytical type in nature through Problem solving**.** This is attained through the assessment component mentioned in course handout.

Course Code: CSE1500	**Course Title: Computational Thinking using Python** **Type of Course: Engineering Science Theory Integrated**	**L-T-P-C**	2	0	0	2
Version No.	1.0					
Course Pre-requisites	•					
Anti-requisites	NIL					
Course Description	The course efficiently introduces fundamental ideas including conditionals, loops, functions, lists, strings, and tuples through some inspiring examples. It then discusses dynamic programming like handling exceptions and file usage. In terms of data structures, the					

		course covers Python dictionaries, classes, and objects for constructing user-defined datatypes like linear and binary search.
Course Object		The objective of the course is to familiarize the learners with the concepts of **Computational Thinking using Python** and attain **Skill Development** through **Participative Learning** techniques.
Course Out Comes		On successful completion of the course the students shall be able to: 1) Describe algorithmic solutions for basic computing issues.. (Understand) 2) Explain data types and operators. (Understand) 3) Demonstrate control structures and Functions. (Apply) 4) Apply the data structures for the given data. (Apply) 5) Demonstrate the file operations. (Apply)
Course Content:		

Module 1	**Computational Thinking And Problem Solving**	Assignment		Programming	**6 Sessions**

Topics:

Fundamentals of Computing– Identification of Computational Problems Algorithms, building blocks of algorithms (statements, state, control flow, functions), notation (pseudo code, flow chart, programming language), algorithmic problem solving, simple strategies for developing algorithms (iteration, recursion). Illustrative problems: find minimum in a list, insert a card in a list of sorted cards, guess an integer number in a range, Towers of Hanoi

Module 2	**Datatypes, Expressions, Statements**	Assignment		Programming	**6 Sessions**

Topics:

Python interpreter and interactive mode,debugging; values and types: int, float, boolean, string , and list; variables, expressions, statements, tuple assignment, precedence of operators, comments; Illustrative programs: exchange the values of two variables, circulate the values of n variables, distance between two points.

Module 3	**Control flow, Functions, Strings**	Assignment		Programming	**6 Sessions**

	Topics: Conditionals:Boolean values and operators, conditional (if), alternative (if else),chained conditional (if-elif-else);Iteration: state, while, for, break, continue, pass; Fruitful functions: return values,parameters, local and global scope, function composition, recursion; Strings: string slices,immutability, string functions and methods, string module; Lists as arrays. Illustrative programs: square root, gcd, exponentiation, sum an array of numbers, linear search, binary search.

Module 4	**Lists, Tuples, Dictionaries**	Assignment		Programming	**6 Sessions**

	Topics: Lists: list operations, list slices, list methods, list loop, mutability, aliasing, cloning lists, list parameters; Tuples: tuple assignment, tuple as return value; Dictionaries: operations and methods; advanced list processing- list comprehension; Illustrative programs: simple sorting, histogram, Students marks statement, Retail bill preparation.

Module 5	**Files**	Assignment		Programming	**6 Sessions**

	Files and exceptions: text files, reading and writing files, format operator; command line arguments, errors and exceptions, handling exceptions, modules, packages; Illustrative programs: word count, copy file, Voter's age validation, Marks range validation (0-100).
	Project work/Assignment:
	2. Assignment 1 on (Module 1 and Module 2) **3. Assignment 2 on (Module 3 and Module 4 & 5)**
	Text Book 1) Paul Deitel and Harvey Deitel, "Python for Programmers", Pearson Education, 1st Edition, 2021 2) Eric Matthes, Python Crash Course,: A Hands-On, Project-Based Introduction to Programming, 3rd Edition, 2023

	References 1.Allen B. Downey, “Think Python: How to Think like a Computer Scientist”, 2nd Edition, O’Reilly Publishers, 2016. 2. Karl Beecher, “Computational Thinking: A Beginner's Guide to Problem Solving and Programming”, 1st Edition, BCS Learning & Development Limited, 2017. **Web Resources** W1. https://onlinecourses.nptel.ac.in/noc20_cs70/preview
	Topics relevant to development of “Employability”: Data structures using python. **Topics relevant to “PROFESSIONAL ETHICS”:** Naming and coding convention for simple programs using python.

Course Code: CSE1510	**Course Title: Database Management Systems** **Type of Course: 1) Program Core & Theory only**	**L-T-P-C**	3	0	0	3
Version No.	1.0					
Course Pre-requisites	Foundational understanding of data types, data structures, basic programming knowledge, familiarity with operating systems and file management. Basic knowledge of set theory, logic, and discrete mathematics to understand relational algebra and query formulation.					
Anti-requisites	NIL					
Course Description	This course introduces the foundational principles of database management systems, including data models, schemas, and architectures. This course provides a solid foundation on the relational model of data and the use of relational algebra. It develops skills in SQL for data definition, manipulation, and control, enabling students to construct and execute complex queries. The course also introduces the concept of object oriented and object relational databases and modern database technologies like **NoSQL.** The also course allows the students to gain insights into data storage structures and indexing					

	strategies for optimizing query performance.
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. Describe the fundamental elements of relational database management systems. [Understand] 2. Examine databases using SQL query processing and Optimization. [Apply] 3. Design simple database systems applying the normalization constraints and demonstrate the database transaction processing, recovery, and security. [Apply] 4. Interpret the concept of advanced databases and its applications. [Apply]

Course Content:

Module 1	**Introduction to Database Modelling and Relational Algebra (Understand)**	Assignment	**Problem Solving**	**10 Sessions**

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 (Apply)**	Assignment	**Programming**	**11 Sessions**

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(Apply)**	Assignment	**Problem Solving**	**12 Sessions**

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 (Apply)**	**Assignment**	**Case Study**	**12 Sessions**

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.

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.

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 Books:

T1. Elmasri R and Navathe S B, “Fundamentals of Database System”, Pearson Publication, 7th Edition, 2018.

T2. RamaKrishna & Gehrke, “Database Management Systems” 3rd Edition, 2018, McGraw-Hill Education.

T3. 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

R1 Avi Silberschatz, Henry F. Korth, S. Sudarshan, “Database System Concepts”, McGraw-Hill ,7th Edition, 2019.

R2 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: CSE1501	**Course Title: Computational Thinking using Python Lab** **Type of Course: Engineering Science Laboratory Integrated**	**L-T-P-C**	0	0	4	2
Version No.	1.0					
Course Pre-requisites	•	•				
Anti-requisites	NIL					
Course Description	The course efficiently introduces fundamental ideas and practical knowledge including control structures, functions, and tuples through hands on sessions. It also introduces dynamic programming like handling exceptions and file operations. The course covers Python dictionaries, classes, and objects for creating user-defined datatypes, such binary search, in terms of data structures.					

Course Object		The objective of the course is to familiarize the learners with the concepts of **Computational Thinking using Python Lab** and attain **Skill Development** through **Experiential Learning** techniques.
Course Out Comes		On successful completion of the course the students shall be able to: 6) Explain algorithms to solve fundamental computational problem. (Understand) 7) Illustrate the use of different data types and operators in Python. (Apply) 8) Demonstrate conditionals, loops, and functions to address problem-solving tasks. (Apply) 9) Utilize appropriate data structures to efficiently manage and process data. (Apply) 10) Perform file handling tasks such as reading, writing, and modifying files in Python.. (Apply)
Course Content:		

Module 1	**Computational Thinking And Problem Solving**	Assignment		Programming	**6 Sessions**

Lab sheet :

Introduction to Python Programming.

Demonstration of Colabs or Jupiter environment.

Demonstrate Input function.

Demonstrate int method.

Demonstrate data types

Demonstrate operators.

Demonstrate simple programs for python environment.

Python program that takes an integer input and calculates the sum of its digits.

Module 2	**Datatypes, Expressions, Statements**	Assignment		Programming	**14 Sessions**

Topics:

Python program to count the number of times a given character appears in a string.

Python program to identify the data types of given variables.

A grocery store needs a billing system, write a python program that calculates the total bill amount based on the price of individual items and their quantities.

A car rental company wants to calculate the mileage (km per liter) for different vehicles based on distance traveled and fuel consumed. Write a Python program to calculate the mileage.

A company wants to calculate the net salary of an employee after deducting tax (10%) and provident fund (5%) from the gross salary. Write a Python program to calculate net salary.

In a student grading system where the final grade depends on whether the student has passed both the written and practical exams. You need to check if the student has passed based on certain conditions:

- **The student must score at least 40 in the written exam.**
- **The student must score at least 50 in the practical exam.**
- **The student must have attended at least 75% of the classes**

Write a Python program to check if a person is eligible to vote. The criteria are:

- **The person must be at least 18 years old.**
- **The person should be a citizen of the country.**

Write a Python program to classify a person into age groups:

- **Child: 0 to 12 years**
- **Teenager: 13 to 19 years**
- **Adult: 20 years and above**

A user authentication system that checks if the user is authorized based on certain conditions, like having a correct password and being over the age of 18. Write a Python program for the above scenario.

In a student registration system, a student must meet certain criteria to be eligible for course registration:

- **The student must have a GPA of 3.0 or above.**

- **The student must have completed the prerequisite course (True/False).**

Module 3	**Control flow, Functions, Strings**	Assignment		Programming	**16 Sessions**

Lab Sheet:

An e-commerce store that offers discounts based on the following criteria:

- **The customer must be a loyal customer (i.e., True).**
- **The total purchase amount must be greater than $100.**

Python program to print the Fibonacci sequence up to n terms

Python program to print the Fibonacci sequence up to n terms using Recursion.

Apply slicing on the given data or dictionary.

Python Programs to create array and print the array.

Python program to check if a given number is an Armstrong number. An Armstrong number for a 3-digit number is one where the sum of the cubes of its digits is equal to the number itself.

The media platform wants to count the number of words in user-submitted posts to enforce character limits or to analyze the length of posts. .

In a bookstore inventory system, You need to implement a feature that checks if a book title is a palindrome. The bookstore wants to offer special discounts for books with titles that are palindromes. You need to create a Python function that reads the book title and determines if it's a palindrome.

In a library management system. The library has a database of books identified by unique numbers (IDs). The library staff wants to apply a special offer to books whose IDs are prime numbers. You need to create a Python program that finds all prime numbers between a given range of book IDs.

In a school management system that stores the marks of students for each subject. You are asked to compute the average marks of a student to evaluate their overall performance. Write a Python program that takes the marks of a student in different subjects and calculates the average.

A small inventory list where you need to search for a specific product ID. Since the list isn't sorted, you can use linear search, which checks each element sequentially until it finds the target. Write a python program to perform linear search.

A sorted list of product IDs and need to quickly find a specific product. Binary search is ideal for this scenario because it efficiently narrows down the search space by repeatedly dividing the list into two halves.

Module 4	**Lists, Tuples, Dictionaries**	Assignment		Programming	**12 Sessions**

<u>Lab Sheet:</u>

Demonstrate List, Tuple and Dictionary.

A supermarket wants to maintain a list of available products and update it when new products arrive or old products are sold out.

A library maintains book records using dictionaries, where the book title is the key and the quantity available is the value.

A school stores student grades in a list, and the teacher wants to see only the top 3 grades.

A restaurant receives online orders in a queue (list) and processes the first 3 orders at a time. Write a Python program to handle orders using list slicing.

A university has course details stored in tuples. The system should extract and display only the course codes. Write Python program to extract course code from tuples.

A fitness tracking app stores a user's daily step count for a week and extracts steps from Monday to Friday. WAP to extract weekly steps using slicing.

A school stores student marks in a list. Write a program to:

- **Find the highest and lowest marks.**
- **Calculate the average marks.**
- **Count how many students scored above 75.**

A company maintains a list of employees' names. Write a program to:

- **Add a new employee to the list.**

- **Remove an employee from the list.**
- **Sort and display all employees in alphabetical order.**

A tuple stores flight details (Flight Number, Destination, Duration). Write a program to:

- **Display all flights.**
- **Find flights with a duration of more than 3 hours.**
- **Access the destination of a specific flight.**

A grocery store stores item details as tuples (Item Name, Price per kg). Write a program to:

- **Calculate the total bill for a customer.**
- **Find the cheapest item.**
- **Sort items by price in ascending order.**

Use Dictionaries: A library stores book records as {Book Title: Copies Available}. Write a program to:

- **Borrow a book (decrease count).**
- **Return a book (increase count).**
- **Display all available books.**

Use List Comprehension: A company stores employee ID numbers. Write a Python program to extract only the even employee IDs from a given list.

Module 5	**Files**	Assignment		Programming	**12 Sessions**

<u>Lab Sheet:</u>

Write a Python program that asks for a voter's age. If the age is below 18, raise an exception "Invalid Age: Must be 18 or older".

Write a Python program that counts the total number of lines, words, and characters in a given text file.

	Write a Python program that reads text file and finds the most repeated word. **Write a program that searches for a word in a file ".txt" and replaces it with another word.** **Write a Python program that copies the content from "source.txt" to "destination.txt". If "destination.txt" does not exist, create it.** **Write a Python program that takes two numbers as command-line arguments and prints their sum.** **Write a Python program that asks for a user's name, age, and marks in three subjects, then formats and displays the result in a structured way. Generate report using string formatting.** **Create a module called "mymath.py" with functions add(a, b), subtract(a, b), and multiply(a, b). write a separate Python script that imports this module and uses these functions.** **Write a Python program that tries to read a file ".txt". If the file is not found, catch the exception and display a message.**
	Project work/Assignment:
	1.Assignment 1 on (Module 1 and Module 2) **5. Assignment 2 on (Module 3 and Module 4 & 5)**
	Text Book 1Paul Deitel and Harvey Deitel, "Python for Programmers", Pearson Education, 1st Edition, 2021 2)Eric Matthes, Python Crash Course,: A Hands-On, Project-Based Introduction to Programming, 3rd Edition, 2023

	References 1.Allen B. Downey, “Think Python: How to Think like a Computer Scientist”, 2nd Edition, O’Reilly Publishers, 2016. 2. Karl Beecher, “Computational Thinking: A Beginner's Guide to Problem Solving and Programming”, 1st Edition, BCS Learning & Development Limited, 2017. **Web Resources** W2. https://onlinecourses.nptel.ac.in/noc20_cs70/preview W3. https://onlinecourses.swayam2.ac.in/cec23_cs02/preview W4. https://www.coursera.org/learn/ai-python-for-beginners
	Topics relevant to development of “Employability”: Data structures using python. **Topics relevant to “PROFESSIONAL ETHICS”:** Naming and coding convention for simple programs using python.

Course Code: CSE1511	**Course Title: Database Management Systems Laboratory** **Type of Course: 1) Laboratory**	**L-T-P-C**	0	0	2	1
Version No.	1.0					
Course Pre-requisites	Foundational understanding of data types, basic programming knowledge, operating systems and file management.					
Anti-requisites	NIL					
Course Description	The Database Management Systems (DBMS) Laboratory is designed to provide students with hands-on experience in database design, implementation, and management using SQL and database management tools such as MySQL. The lab complements theoretical concepts learned in database courses by allowing students to practice database creation, querying, and optimization techniques. The DBMS Lab enables students to develop					

	industry-relevant skills in database management, preparing them for careers in **software development, data engineering, and database administration**.
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 OutComes	On successful completion of the course the students shall be able to: 5. Demonstrate the database concepts, practice, and SQL queries. [Apply] 6. Design and implement database schemas while applying normalization techniques to optimize structure. [Apply]] 7. Develop and implement stored procedures, triggers, and views for automation and efficiency. [Apply] 8. To Design and build database applications for real world problems. [Apply]

Course Content:

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]

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

Level 1: Implement Oracle Functions and Triggers in Oracle on Employee database.

Level 2: Determine the requirement and construct Functions and Triggers. [Supply chain Database]

Labsheet-6 [4 Practical Sessions]

Experiment No. 8: [2 Sessions]

8. To implement the concept of forms and reports.

Level 1: Implement the concept of forms and reports.

Level 2: Examine the schema relationship.

Experiment No. 9: [2 Sessions]

9. Create the database using the given schema. (Flight Management)

Level 1: Implement a relational database based on the provided schema for the Flight Management system, including the creation of tables, relationships, and constraints.
Level 2: Demonstrate schema relationships by defining primary and foreign keys to ensure data integrity within the Flight Management database.

Labsheet-7 [4 Practical Sessions]
Experiment No. 10: [2 Sessions]
10. Create the database using the given schema. (Company database)
Level 1: Implement the database schema by defining tables, relationships, and constraints according to the given Company Database schema.
Level 2: Demonstrate the schema's relationships and data integrity by creating and linking tables as per the specified requirements.
Experiment No. 11: [2 Sessions]
11. Create the database using the given schema. (Student Library)
Level 1: Implement forms and reports based on the provided Student Library database schema, ensuring effective data entry and reporting mechanisms.
Level 2: Demonstrate the schema relationships within the Student Library database, demonstrating how these relationships influence the creation and functionality of forms and reports.

Labsheet-8 [1 Sessions]
12. 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.

Course Code: CBD2000
Course Title: Data Structures and Algorithms
L:T:P:C – 3:0:0:3

Course Description
This course provides an in-depth understanding of fundamental data structures and algorithms used in computer science. It covers linear and non-linear data structures, algorithm design paradigms, and performance analysis. The focus is on problem-solving, complexity evaluation, and efficient implementation techniques.

Course Objectives
— Introduce fundamental data structures and their applications
— Develop skills in algorithm design, analysis, and complexity evaluation

— Explore sorting, searching, and graph algorithms
— Enable students to choose appropriate data structures for various problems

---

Course Outcomes
CO1 (Understand): Explain fundamental data structures and their operations
CO2 (Analyze): Evaluate algorithmic complexity and space-time trade-offs
CO3 (Apply): Apply linear and non-linear data structures for problem solving
CO4 (Apply): Implement searching, sorting, and traversal algorithms effectively

---

Course Content (45 Hours Total)

Module 1: Introduction and Linear Data Structures – 12 Sessions
Abstract Data Types (ADT), Arrays, Strings, Linked Lists (Singly, Doubly, Circular), Stacks and Queues, Applications of Stack and Queue (Expression evaluation, Recursion)
Module 2: Non-linear Data Structures – 11 Sessions
Trees: Binary Trees, BST, Tree Traversals (Inorder, Preorder, Postorder), AVL Trees, Heaps (Min/Max), Priority Queues
Module 3: Graphs and Applications – 11 Sessions
Graph Representations (Adjacency Matrix/List), BFS and DFS, Dijkstra's Algorithm, Minimum Spanning Trees (Prim's and Kruskal's), Topological Sort
Module 4: Searching, Sorting and Algorithm Design – 11 Sessions
Linear and Binary Search, Sorting: Bubble, Selection, Insertion, Merge Sort, Quick Sort, Hashing Techniques, Algorithm Complexity (Big O, Ω, Θ)
Textbooks
T1: Ellis Horowitz, Sartaj Sahni, *Fundamentals of Data Structures in C*, University Press
T2: Mark Allen Weiss, *Data Structures and Algorithm Analysis in C/C++*, Pearson Education

---

Reference Books
R1: Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, *Data Structures and Algorithms*, Pearson
R2: Robert Lafore, *Data Structures and Algorithms in C++*, Sams Publishing
R3: Narasimha Karumanchi, *Data Structures and Algorithms Made Easy*, CareerMonk
R4: Thomas H. Cormen et al., *Introduction to Algorithms*, MIT Press

---

Web Resources
W1: https://www.geeksforgeeks.org
W2: https://visualgo.net
W3: https://leetcode.com

W4: https://www.tutorialspoint.com/data_structures_algorithms
W5: https://ocw.mit.edu

Course Code: CBD2001
Course Title: Data Structures and Algorithms Lab
L:T:P:C – 0-0-4-2

Course Description
This course provides an in-depth understanding of fundamental data structures and algorithms used in computer science. It covers linear and non-linear data structures, algorithm design paradigms, and performance analysis. The focus is on problem-solving, complexity evaluation, and efficient implementation techniques.

Course Objectives
— Introduce fundamental data structures and their applications
— Develop skills in algorithm design, analysis, and complexity evaluation
— Explore sorting, searching, and graph algorithms
— Enable students to choose appropriate data structures for various problems

---

Course Outcomes
CO1 (Understand): Explain fundamental data structures and their operations
CO2 (Analyze): Evaluate algorithmic complexity and space-time trade-offs
CO3 (Apply): Apply linear and non-linear data structures for problem solving
CO4 (Apply): Implement searching, sorting, and traversal algorithms effectively

---

Course Content (45 Hours Total)

Module 1: Introduction and Linear Data Structures – 12 Sessions
Abstract Data Types (ADT), Arrays, Strings, Linked Lists (Singly, Doubly, Circular), Stacks and Queues, Applications of Stack and Queue (Expression evaluation, Recursion)

Module 2: Non-linear Data Structures – 11 Sessions
Trees: Binary Trees, BST, Tree Traversals (Inorder, Preorder, Postorder), AVL Trees, Heaps (Min/Max), Priority Queues

Module 3: Graphs and Applications – 11 Sessions
Graph Representations (Adjacency Matrix/List), BFS and DFS, Dijkstra's Algorithm, Minimum Spanning Trees (Prim's and Kruskal's), Topological Sort

Module 4: Searching, Sorting and Algorithm Design – 11 Sessions
Linear and Binary Search, Sorting: Bubble, Selection, Insertion, Merge Sort, Quick Sort, Hashing Techniques, Algorithm Complexity (Big O, Ω, Θ)

	Lab Experiment Title	Key Concepts / Structures
Week		
1	**Implement array operations: insertion, deletion, traversal**	**Arrays**
2	**Implement linked list operations (Singly Linked List)**	**Pointers, Nodes, Memory Allocation**
3	**Implement Doubly Linked List and Circular Linked List**	**Advanced Linked Lists**
4	**Stack implementation using arrays and linked list**	**Stack ADT, LIFO**
5	**Queue and Circular Queue implementation**	**Queue ADT, FIFO**
6	**Infix to Postfix conversion and evaluation of postfix expressions**	**Stack-based expression evaluation**
7	**Implementation of Binary Search Tree (BST)**	**Tree Traversals, Insertion, Deletion**
8	**Implement AVL Tree with insertion and rotations**	**Balanced BST, Rotations**
9	**Depth-First Search (DFS) and Breadth-First Search (BFS) on graphs**	**Graph Traversal, Recursion, Queue**
10	**Implement Dijkstra's Algorithm for shortest path**	**Graphs, Priority Queue**
11	**Implement Merge Sort and Quick Sort**	**Divide and Conquer, Recursion**
12	**Implement Heap and perform Heap Sort**	**Max Heap, Min Heap**
13	**Implement Hash Table with collision resolution (Linear/Quadratic Probing, Chaining)**	**Hashing, Collision Handling**
14	**Implement Kruskal's and Prim's algorithm for Minimum Spanning Tree**	**Greedy Algorithms, Disjoint Sets**

15	**Mini Project: Design and implement a data structure-based application (group-wise)**	**Integration of Multiple DS & Algorithms**

Languages/Tools Recommended:

- **C/C++, Java, or Python (based on department policy)**

Textbooks

T1: Ellis Horowitz, Sartaj Sahni, *Fundamentals of Data Structures in C*, University Press
T2: Mark Allen Weiss, *Data Structures and Algorithm Analysis in C/C++*, Pearson Education

Reference Books

R1: Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, *Data Structures and Algorithms*, Pearson
R2: Robert Lafore, *Data Structures and Algorithms in C++*, Sams Publishing
R3: Narasimha Karumanchi, *Data Structures and Algorithms Made Easy*, CareerMonk
R4: Thomas H. Cormen et al., *Introduction to Algorithms*, MIT Press

Web Resources

W1: https://www.geeksforgeeks.org
W2: https://visualgo.net
W3: https://leetcode.com
W4: https://www.tutorialspoint.com/data_structures_algorithms
W5: https://ocw.mit.edu

Course Code: CBD2500
Course Title: Data Visualization and Reporting
L:T:P:C – 3:0:0:3

Course Description

This course introduces techniques and tools for effective data visualization and reporting. It covers principles of visual encoding, dashboard design, storytelling with data, and the use of modern visualization tools and libraries. The course helps learners convert complex datasets into intuitive, actionable insights for decision-making.

Course Objectives

— Understand the principles and theory of data visualization
— Explore various types of charts, plots, and dashboards for representing different

data types
— Learn to use tools and libraries for building interactive visualizations
— Develop effective and ethical data stories and reports

Course Outcomes
CO1 (Understand): Explain the importance, types, and design principles of data visualization
CO2 (Analyze): Select appropriate visualization techniques for given data types and context
CO3 (Apply): Create interactive visualizations and dashboards using modern tools
CO4 (Apply): Generate meaningful reports and tell compelling stories from data

---

Course Content (45 Hours Total)

Module 1: Introduction to Data Visualization – 11 Sessions
Definition and scope of data visualization, Importance of visualization in analytics, Visual encoding (position, color, size), Types of data (categorical, numerical, time-series), Chart selection guidelines, Exploratory vs explanatory visualization
Module 2: Visualization Tools and Libraries – 11 Sessions
Overview of tools: Tableau, Power BI, Google Data Studio, D3.js, Matplotlib, Seaborn, Plotly, Altair, Setting up data pipelines and basic visualization workflows
Module 3: Designing Dashboards and Interactive Visuals – 11 Sessions
Dashboard design principles, Filters, Drill-downs, Linking charts, User experience in dashboards, Responsive design, Real-time data visualization, Use cases in business and IoT
Module 4: Storytelling and Reporting with Data – 12 Sessions
Data-driven storytelling, Narrative flow, Ethical considerations in visualization, Report generation (PDF, Web), Use of annotations, infographics, and presentations, Automating visual reports
Textbooks
T1: Cole Nussbaumer Knaflic, *Storytelling with Data: A Data Visualization Guide for Business Professionals*, Wiley
T2: Nathan Yau, *Data Points: Visualization That Means Something*, Wiley

---

Reference Books
R1: Alberto Cairo, *The Truthful Art: Data, Charts, and Maps for Communication*, New Riders
R2: Ben Fry, *Visualizing Data: Exploring and Explaining Data with the Processing Environment*, O'Reilly
R3: Andy Kirk, *Data Visualization: A Handbook for Data Driven Design*, Sage
R4: Claus O. Wilke, *Fundamentals of Data Visualization*, O'Reilly

---

Web Resources
W1: https://www.tableau.com/learn
W2: https://powerbi.microsoft.com
W3: https://seaborn.pydata.org
W4: https://d3js.org
W5: https://datavizcatalogue.com

Course Code: CBD2501
Course Title: Data Visualization and Reporting
L:T:P:C – 0-0-2-1

Course Description
This course introduces techniques and tools for effective data visualization and reporting. It covers principles of visual encoding, dashboard design, storytelling with data, and the use of modern visualization tools and libraries. The course helps learners convert complex datasets into intuitive, actionable insights for decision-making.

Course Objectives
— Understand the principles and theory of data visualization
— Explore various types of charts, plots, and dashboards for representing different data types
— Learn to use tools and libraries for building interactive visualizations
— Develop effective and ethical data stories and reports

Course Outcomes
CO1 (Understand): Explain the importance, types, and design principles of data visualization
CO2 (Analyze): Select appropriate visualization techniques for given data types and context
CO3 (Apply): Create interactive visualizations and dashboards using modern tools
CO4 (Apply): Generate meaningful reports and tell compelling stories from data

---

Course Content (45 Hours Total)

Module 1: Introduction to Data Visualization – 11 Sessions
Definition and scope of data visualization, Importance of visualization in analytics, Visual encoding (position, color, size), Types of data (categorical, numerical, time-series), Chart selection guidelines, Exploratory vs explanatory visualization

Module 2: Visualization Tools and Libraries – 11 Sessions
Overview of tools: Tableau, Power BI, Google Data Studio, D3.js, Matplotlib, Seaborn, Plotly, Altair, Setting up data pipelines and basic visualization workflows

Module 3: Designing Dashboards and Interactive Visuals – 11 Sessions
Dashboard design principles, Filters, Drill-downs, Linking charts, User experience in dashboards, Responsive design, Real-time data visualization, Use cases in business and IoT

Module 4: Storytelling and Reporting with Data – 12 Sessions
Data-driven storytelling, Narrative flow, Ethical considerations in visualization, Report generation (PDF, Web), Use of annotations, infographics, and presentations, Automating visual reports

List of Experiments

Week	Lab Experiment Title	Tools / Technologies
1	Introduction to Data Visualization and Tool Setup	Tableau / Power BI / Python
2	Visualizing Categorical and Numerical Data using Basic Charts	Tableau / Matplotlib
3	Creating Histograms, Box Plots, and Heatmaps	Seaborn / Power BI
4	Building Multi-dimensional Charts (Bubble, Pair Plot, Violin Plot)	Seaborn / Plotly
5	Time Series Visualization with Line Charts and Area Charts	Tableau / Plotly / Power BI
6	Dashboard Design: Adding Filters, Tooltips, and Interactions	Tableau / Power BI
7	Comparative Analysis using Bar Charts and Stacked Graphs	Power BI / Seaborn
8	Geo-visualization using Maps for Regional Data	Tableau / Google Data Studio
9	Real-time Data Visualization using APIs	Plotly / Streamlit
10	Storytelling with Data: Narrating Data Insights	Google Data Studio / Tableau
11	Generating Reports from Dashboards (PDF, HTML exports)	Power BI / Tableau / Jupyter
12	Creating Interactive Visualizations with Plotly	Plotly Express
13	Automating Visualization Pipelines with Python	Pandas + Matplotlib + ReportLab
14	Case Study: Sales or Healthcare Dashboard (Group-wise)	Any platform (student choice)
15	Mini Project Presentation: Domain-based Visual Reporting Application	Multiple Tools

🛠 **Recommended Tools & Platforms:**

- **Tableau Public**
- **Microsoft Power BI Desktop**
- **Python (Jupyter Notebook with Matplotlib, Seaborn, Plotly, Pandas)**

Textbooks

T1: Cole Nussbaumer Knaflic, *Storytelling with Data: A Data Visualization Guide for Business Professionals*, Wiley
T2: Nathan Yau, *Data Points: Visualization That Means Something*, Wiley

Reference Books

R1: Alberto Cairo, *The Truthful Art: Data, Charts, and Maps for Communication*, New Riders
R2: Ben Fry, *Visualizing Data: Exploring and Explaining Data with the Processing Environment*, O'Reilly
R3: Andy Kirk, *Data Visualization: A Handbook for Data Driven Design*, Sage
R4: Claus O. Wilke, *Fundamentals of Data Visualization*, O'Reilly

Web Resources

W1: https://www.tableau.com/learn
W2: https://powerbi.microsoft.com
W3: https://seaborn.pydata.org
W4: https://d3js.org
W5: https://datavizcatalogue.com

Course Code: CBD2502
Course Title: Data Mining and Predictive Analytics
L:T:P:C – 3:0:0:3

Course Description
This course provides an introduction to data mining concepts and predictive modeling techniques. It explores data preprocessing, classification, clustering, association rule mining, and model evaluation, along with real-world applications of predictive analytics using tools like Python, R, and data mining platforms.

Course Objectives
— Understand the concepts, techniques, and algorithms in data mining
— Explore various predictive modeling and classification techniques
— Develop skills to build, evaluate, and interpret predictive models
— Apply data mining techniques to extract patterns from large datasets

---

Course Outcomes
CO1 (Understand): Explain fundamental concepts of data mining and predictive analytics
CO2 (Analyze): Evaluate and compare algorithms for classification, clustering, and association
CO3 (Apply): Develop predictive models using machine learning techniques
CO4 (Apply): Use data mining tools to perform pattern discovery and decision making

---

Course Content (45 Hours Total)

Module 1: Introduction to Data Mining and Preprocessing – 11 Sessions
Overview of data mining, KDD process, Data types and formats, Data cleaning, integration, transformation, normalization, Data reduction, Feature selection
Module 2: Classification and Prediction Techniques – 12 Sessions
Classification vs prediction, Decision trees, Naïve Bayes, k-NN, Logistic Regression, Support Vector Machines (SVM), Model accuracy, Confusion matrix, ROC, Precision-Recall
Module 3: Clustering and Association Rule Mining – 11 Sessions
Partitioning methods (k-Means), Hierarchical clustering, DBSCAN, Apriori algorithm, FP-Growth, Market basket analysis, Interestingness measures (support, confidence, lift)
Module 4: Predictive Analytics Applications – 11 Sessions
Predictive analytics in business, health, and finance, Case studies, Introduction to tools (Orange, RapidMiner, Weka, Python libraries), Model deployment and interpretation

Textbooks
T1: Jiawei Han, Micheline Kamber, Jian Pei, *Data Mining: Concepts and Techniques*, Elsevier, 3rd Edition
T2: Trevor Hastie, Robert Tibshirani, Jerome Friedman, *The Elements of Statistical Learning*, Springer

---

Reference Books
R1: Pang-Ning Tan, Michael Steinbach, Vipin Kumar, *Introduction to Data Mining*, Pearson
R2: Ian H. Witten, Eibe Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, Elsevier
R3: Galit Shmueli, Nitin R. Patel, Peter C. Bruce, *Data Mining for Business Analytics*, Wiley
R4: Daniel T. Larose, *Discovering Knowledge in Data: An Introduction to Data Mining*, Wiley

---

Web Resources
W1: https://scikit-learn.org
W2: https://www.kaggle.com/learn
W3: https://www.datasciencecentral.com
W4: https://www.r-bloggers.com
W5: https://towardsdatascience.com

Course Code: CBD2503
Course Title: Data Mining and Predictive Analytics Lab
L:T:P:C – 0:0:2:1

Course Description
This course provides an introduction to data mining concepts and predictive modeling techniques. It explores data preprocessing, classification, clustering, association rule mining, and model evaluation, along with real-world applications of predictive analytics using tools like Python, R, and data mining platforms.

Course Objectives
— Understand the concepts, techniques, and algorithms in data mining
— Explore various predictive modeling and classification techniques
— Develop skills to build, evaluate, and interpret predictive models
— Apply data mining techniques to extract patterns from large datasets

---

Course Outcomes
CO1 (Understand): Explain fundamental concepts of data mining and predictive analytics
CO2 (Analyze): Evaluate and compare algorithms for classification, clustering, and association
CO3 (Apply): Develop predictive models using machine learning techniques
CO4 (Apply): Use data mining tools to perform pattern discovery and decision making

Course Content (45 Hours Total)

Module 1: Introduction to Data Mining and Preprocessing – 11 Sessions
Overview of data mining, KDD process, Data types and formats, Data cleaning, integration, transformation, normalization, Data reduction, Feature selection
Module 2: Classification and Prediction Techniques – 12 Sessions
Classification vs prediction, Decision trees, Naïve Bayes, k-NN, Logistic Regression, Support Vector Machines (SVM), Model accuracy, Confusion matrix, ROC, Precision-Recall
Module 3: Clustering and Association Rule Mining – 11 Sessions
Partitioning methods (k-Means), Hierarchical clustering, DBSCAN, Apriori algorithm, FP-Growth, Market basket analysis, Interestingness measures (support, confidence, lift)
Module 4: Predictive Analytics Applications – 11 Sessions
Predictive analytics in business, health, and finance, Case studies, Introduction to tools (Orange, RapidMiner, Weka, Python libraries), Model deployment and interpretation

List of Experiments

Week	Lab Experiment Title	Tools / Technologies
1	Introduction to Data Mining Tools and Datasets	Weka / Orange / Jupyter (Python)
2	Data Preprocessing: Cleaning, Normalization, Encoding	Python (pandas, sklearn)
3	Data Visualization and Exploration	seaborn / matplotlib
4	Implement Decision Tree Classification	Python (scikit-learn) / Weka
5	Implement Naïve Bayes Classifier	Python (scikit-learn)
6	Implement k-Nearest Neighbors (k-NN)	Python (scikit-learn)
7	Logistic Regression for Binary and Multiclass Prediction	Python (scikit-learn) / R
8	Model Evaluation: Confusion Matrix, Accuracy, Precision, Recall, F1-Score	Python (metrics module)
9	Clustering with k-Means and Visualizations	Python (scikit-learn, seaborn)
10	Hierarchical Clustering and Dendrograms	scipy.cluster.hierarchy
11	Market Basket Analysis using Apriori Algorithm	Weka / mlxtend (Python)
12	FP-Growth Algorithm for Frequent Pattern Mining	Orange / Python (pyfpgrowth)
13	Mini Project Part 1: Real-world classification or prediction dataset analysis	Any platform
14	Mini Project Part 2: Model building and performance reporting	Python / RapidMiner / Orange
15	Mini Project Presentation and Demonstration	Any suitable tools

🛠 **Recommended Tools & Libraries**

- **Languages: Python (pandas, scikit-learn, matplotlib, seaborn, mlxtend)**
- **Tools: Weka, Orange, RapidMiner, Jupyter Notebook**
- **Optional: R (caret, ggplot2), Tableau for visualization**

Textbooks

T1: Jiawei Han, Micheline Kamber, Jian Pei, *Data Mining: Concepts and Techniques*, Elsevier, 3rd Edition
T2: Trevor Hastie, Robert Tibshirani, Jerome Friedman, *The Elements of Statistical Learning*, Springer

Reference Books

R1: Pang-Ning Tan, Michael Steinbach, Vipin Kumar, *Introduction to Data Mining*, Pearson
R2: Ian H. Witten, Eibe Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, Elsevier
R3: Galit Shmueli, Nitin R. Patel, Peter C. Bruce, *Data Mining for Business Analytics*, Wiley
R4: Daniel T. Larose, *Discovering Knowledge in Data: An Introduction to Data Mining*, Wiley

Web Resources

W1: https://scikit-learn.org
W2: https://www.kaggle.com/learn
W3: https://www.datasciencecentral.com
W4: https://www.r-bloggers.com
W5: https://towardsdatascience.com

Course Code: CBD2504

Course Title: Fundamentals of Data Analytics

L:T:P:C – 3:0:0:3

Prerequisite:CBD1700 Introduction to Big Data

Course Description

This course introduces the foundational concepts of data analytics, including data understanding, preprocessing, exploratory data analysis, and the basics of statistical inference. It provides the skills necessary to derive insights from data and prepares students for further study in advanced analytics, machine learning, and data science.

Course Objectives

— Understand the life cycle and processes involved in data analytics
— Explore various data types, sources, and preprocessing methods
— Apply exploratory and statistical analysis techniques to interpret data
— Use basic visualization techniques to communicate findings

Course Outcomes

CO1 : Describe the stages of the data analytics lifecycle and types of data**(Understand)**
CO2 : Perform exploratory data analysis using descriptive statistics**(Analyse)**
CO3 : Apply data preprocessing and transformation techniques**(Apply)**
CO4: Use visual analytics to draw meaningful insights from datasets**(Apply)**

Course Content (45 Hours Total)

Module 1: Introduction to Data Analytics – 11 Sessions (Understand)
Data analytics lifecycle, Types of data (structured, semi-structured, unstructured), Data sources, Characteristics of good data, Use cases across industries, Role of analytics in decision making

---

Module 2: Data Preprocessing and Cleaning – 11 Sessions (Analyse)
Handling missing values, Outlier detection and treatment, Data transformation (scaling, encoding), Feature engineering, Data integration and aggregation, Data sampling

---

Module 3: Exploratory Data Analysis (EDA) – 11 Sessions (Apply)
Univariate and multivariate analysis, Measures of central tendency and dispersion, Correlation, Cross-tabulation, Pivot tables, Data distribution and patterns

---

Module 4: Introduction to Statistical Analysis and Visualization – 12 Sessions (Apply)
Hypothesis testing basics, Confidence intervals, p-values, Visual analytics: histograms, scatter plots, box plots, line charts, Introduction to tools: Excel, Python, R

Textbooks
T1: Cathy O'Neil and Rachel Schutt, *Doing Data Science*, O'Reilly Media
T2: Vignesh Prajapati, *Big Data Analytics with R and Hadoop*, Packt Publishing

---

Reference Books
R1: Anil Maheshwari, *Data Analytics*, McGraw Hill
R2: John W. Foreman, *Data Smart*, Wiley
R3: Alberto Cairo, *The Functional Art: An Introduction to Information Graphics and Visualization*, New Riders
R4: Dean Abbott, *Applied Predictive Analytics*, Wiley

Web Resources
W1: https://www.kaggle.com/learn
W2: https://www.datacamp.com
W3: https://pandas.pydata.org
W4: https://www.rstudio.com
W5: https://www.tidyverse.org

Course Code: CBD2505
Course Title: Fundamentals of Data Analytics lab
L:T:P:C – 0-0-4-2

Course Description
This course introduces the foundational concepts of data analytics, including data understanding, preprocessing, exploratory data analysis, and the basics of statistical inference. It provides the skills necessary to derive insights from data and prepares students for further study in advanced analytics, machine learning, and data science.

Course Objectives
— Understand the life cycle and processes involved in data analytics
— Explore various data types, sources, and preprocessing methods
— Apply exploratory and statistical analysis techniques to interpret data
— Use basic visualization techniques to communicate findings

Course Outcomes
CO1 (Understand): Describe the stages of the data analytics lifecycle and types of data
CO2 (Analyze): Perform exploratory data analysis using descriptive statistics
CO3 (Apply): Apply data preprocessing and transformation techniques
CO4 (Apply): Use visual analytics to draw meaningful insights from datasets

---

Course Content (45 Hours Total)

Module 1: Introduction to Data Analytics – 11 Sessions
Data analytics lifecycle, Types of data (structured, semi-structured, unstructured), Data sources, Characteristics of good data, Use cases across industries, Role of analytics in decision making
Module 2: Data Preprocessing and Cleaning – 11 Sessions
Handling missing values, Outlier detection and treatment, Data transformation (scaling, encoding), Feature engineering, Data integration and aggregation, Data sampling

Module 3: Exploratory Data Analysis (EDA) – 11 Sessions
Univariate and multivariate analysis, Measures of central tendency and dispersion, Correlation, Cross-tabulation, Pivot tables, Data distribution and patterns
Module 4: Introduction to Statistical Analysis and Visualization – 12 Sessions
Hypothesis testing basics, Confidence intervals, p-values, Visual analytics: histograms, scatter plots, box plots, line charts, Introduction to tools: Excel, Python, R

List of Experiments

Week	Lab Experiment Title	Tools / Technologies
1	Introduction to the analytics environment and tools	Excel / Python / R
2	Importing and exploring datasets	Python (pandas), R (readr, dplyr)
3	Handling missing data and duplicate values	Python / Excel / R
4	Data cleaning and formatting (date parsing, string operations)	Python (pandas), R
5	Exploratory Data Analysis: Summary statistics and distributions	Python (describe, groupby)
6	Univariate analysis using histograms, bar charts, pie charts	seaborn / matplotlib / Excel
7	Bivariate analysis: Scatter plots, correlation matrices	seaborn.pairplot(), heatmap()
8	Box plots and outlier detection	seaborn / matplotlib / Excel
9	Feature scaling: Normalization and standardization	sklearn.preprocessing / R
10	Encoding categorical variables: One-hot, label encoding	pandas / sklearn
11	Pivot tables and cross-tabulations for summary analysis	Excel / pandas.pivot_table()
12	Basic statistical analysis: mean, median, mode, variance, standard deviation	Python / Excel formulas
13	Hypothesis testing basics (t-test, chi-square test)	SciPy / R
14	Mini Project Part 1: Domain-based EDA and data prep	Any suitable tool
15	Mini Project Part 2: Visual storytelling and reporting with insights	Excel / Tableau / Python

Textbooks
T1: Cathy O'Neil and Rachel Schutt, *Doing Data Science*, O'Reilly Media
T2: Vignesh Prajapati, *Big Data Analytics with R and Hadoop*, Packt Publishing

Reference Books
R1: Anil Maheshwari, *Data Analytics*, McGraw Hill
R2: John W. Foreman, *Data Smart*, Wiley
R3: Alberto Cairo, *The Functional Art: An Introduction to Information Graphics and Visualization*, New Riders
R4: Dean Abbott, *Applied Predictive Analytics*, Wiley

Web Resources

W1: https://www.kaggle.com/learn
W2: https://www.datacamp.com
W3: https://pandas.pydata.org
W4: https://www.rstudio.com
W5: https://www.tidyverse.org

Course Code: **CSE2000**	**Course Title: Software Design and Development** **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		**10 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, Analysis and Design** **(Comprehension level)**	Assignment	Development of SRS documents for a given scenario	**12 Hours**
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		**10 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	**13 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

R1. Roger S. Pressman, "Software Engineering – A Practitioner's Approach", VII Edition, Graw-Hill, 2017.

B2. Bob Hughes, Mike Cotterell, Rajib Mall, "Software Project Management", VI Edition, Graw-Hill, 2018.

References

1. Rajib Mall, "Fundamentals of Software Engineering", VI Edition, PHI learning private limited, 2015.
2. Ian Sommerville, "Software Engineering", IX Edition, Pearson Education Asia, 2011.
3. Agile 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: CBD2506
Course Title: Cloud Computing for Big Data
L:T:P:C – 3:0:0:3

Course Description

This course explores the integration of cloud computing and big data technologies. It introduces cloud service models, deployment models, storage systems, and big data processing frameworks hosted in cloud environments. The course equips students with knowledge to manage, process, and analyze large-scale data using cloud-based tools.

Course Objectives

— Understand the fundamentals and architecture of cloud computing
— Explore cloud service models and deployment types relevant to big data
— Learn cloud storage mechanisms and virtualization concepts
— Apply big data analytics platforms like Hadoop and Spark in the cloud

Course Outcomes

CO1 (Understand): Describe cloud computing architecture and service models
CO2 (Analyze): Compare deployment models and technologies for big data in the cloud
CO3 (Apply): Use cloud storage, virtualization, and distributed file systems
CO4 (Apply): Deploy and manage big data processing frameworks on cloud platforms

Course Content (45 Hours Total)

Module 1: Introduction to Cloud Computing – 11 Sessions

Cloud architecture, Cloud service models (IaaS, PaaS, SaaS), Deployment models (Public, Private, Hybrid), Cloud benefits and challenges, Service-level agreements (SLAs)

Module 2: Cloud Infrastructure and Virtualization – 11 Sessions

Virtualization types: hardware, OS, storage, and network virtualization, Hypervisors, Containerization (Docker), Cloud providers (AWS, Azure, GCP), Pricing models

Module 3: Cloud Storage and Data Management – 11 Sessions

Cloud storage types (object, block, file), Storage services (Amazon S3, Google Cloud Storage), Distributed File Systems (HDFS), Data ingestion tools (Flume, Sqoop), Data lake concepts

Module 4: Big Data Analytics on Cloud – 12 Sessions

Cloud-based big data platforms (Amazon EMR, Dataproc, Azure HDInsight), Hadoop and Spark in cloud environments, Cloud-native analytics (BigQuery, AWS Athena), Workflow orchestration (Apache Airflow)

Textbooks

T1: Rajkumar Buyya et al., *Mastering Cloud Computing*, McGraw Hill Education
T2: Thomas Erl et al., *Cloud Computing: Concepts, Technology & Architecture*, Prentice Hall

Reference Books

R1: Gautam Shroff, *Enterprise Cloud Computing*, Cambridge University Press
R2: Toby Velte et al., *Cloud Computing: A Practical Approach*, McGraw Hill
R3: Michael Miller, *Cloud Computing: Web-Based Applications That Change the Way You Work*, Que Publishing
R4: Dan C. Marinescu, *Cloud Computing: Theory and Practice*, Morgan Kaufmann

Web Resources

W1: https://aws.amazon.com/big-data/
W2: https://cloud.google.com/bigquery
W3: https://azure.microsoft.com/en-us/products/hdinsight/
W4: https://docs.docker.com
W5: https://www.databricks.com

Course Code: CBD2507
Course Title: Cloud Computing for Big Data Lab
L:T:P:C – 0-0-2-1

Course Description
This course explores the integration of cloud computing and big data technologies. It introduces cloud service models, deployment models, storage systems, and big data processing frameworks hosted in cloud environments. The course equips students with knowledge to manage, process, and analyze large-scale data using cloud-based tools.

Course Objectives
— Understand the fundamentals and architecture of cloud computing
— Explore cloud service models and deployment types relevant to big data
— Learn cloud storage mechanisms and virtualization concepts
— Apply big data analytics platforms like Hadoop and Spark in the cloud

---

Course Outcomes
CO1 (Understand): Describe cloud computing architecture and service models
CO2 (Analyze): Compare deployment models and technologies for big data in the cloud
CO3 (Apply): Use cloud storage, virtualization, and distributed file systems
CO4 (Apply): Deploy and manage big data processing frameworks on cloud platforms

---

Course Content (45 Hours Total)

Module 1: Introduction to Cloud Computing – 11 Sessions
Cloud architecture, Cloud service models (IaaS, PaaS, SaaS), Deployment models (Public, Private, Hybrid), Cloud benefits and challenges, Service-level agreements (SLAs)

Module 2: Cloud Infrastructure and Virtualization – 11 Sessions
Virtualization types: hardware, OS, storage, and network virtualization, Hypervisors, Containerization (Docker), Cloud providers (AWS, Azure, GCP), Pricing models

---

Module 3: Cloud Storage and Data Management – 11 Sessions
Cloud storage types (object, block, file), Storage services (Amazon S3, Google Cloud Storage), Distributed File Systems (HDFS), Data ingestion tools (Flume, Sqoop), Data lake concepts

---

Module 4: Big Data Analytics on Cloud – 12 Sessions
Cloud-based big data platforms (Amazon EMR, Dataproc, Azure HDInsight), Hadoop and Spark in cloud environments, Cloud-native analytics (BigQuery, AWS Athena), Workflow orchestration (Apache Airflow)

List of Experiments

Week	Lab Experiment Title	Tools / Technologies
1	**Introduction to cloud platforms and account setup (AWS / GCP / Azure)**	**AWS Educate / GCP Free Tier / Azure for Students**
2	**Launching and managing virtual machines on the cloud**	**AWS EC2 / Google Compute Engine**

3	**Installing and configuring Hadoop on cloud VMs**	**Hadoop on Ubuntu VM**
4	**Working with HDFS: file operations and block management**	**HDFS Commands**
5	**MapReduce Programming: Word Count and other simple jobs**	**Java / Python / Hadoop**
6	**Introduction to Spark and RDD operations**	**Apache Spark (PySpark)**
7	**DataFrame operations and aggregations using PySpark**	**PySpark (Jupyter Notebook / VS Code)**
8	**Working with Amazon S3 / Google Cloud Storage**	**S3 CLI / GCP Storage Console**
9	**Data ingestion using Sqoop and Flume**	**Apache Sqoop / Apache Flume**
10	**Launching Hadoop Cluster using Amazon EMR / Google Dataproc**	**AWS EMR / GCP Dataproc**
11	**Running Spark jobs on the cloud**	**PySpark / Scala (cloud notebook or CLI)**
12	**Introduction to containers: Docker installation and basic container creation**	**Docker CLI / Docker Desktop**
13	**Building and deploying a containerized Spark job**	**Docker + Spark**
14	**Mini Project Part 1 – Use case implementation (e.g., log analysis, e-commerce data pipeline)**	**Cloud VM + HDFS/Spark**
15	**Mini Project Part 2 – Data processing, analytics, and visualization**	**Spark + Cloud Dashboard or Power BI**

🛠 **Recommended Tools & Platforms**

- **Cloud Platforms: AWS, GCP, Microsoft Azure**
- **Big Data Tools: Hadoop, Spark, HDFS, Sqoop, Flume**
- **Languages: Python, Java, Scala**

Textbooks

T1: Rajkumar Buyya et al., *Mastering Cloud Computing*, McGraw Hill Education
T2: Thomas Erl et al., *Cloud Computing: Concepts, Technology & Architecture*, Prentice Hall

---

Reference Books

R1: Gautam Shroff, *Enterprise Cloud Computing*, Cambridge University Press
R2: Toby Velte et al., *Cloud Computing: A Practical Approach*, McGraw Hill
R3: Michael Miller, *Cloud Computing: Web-Based Applications That Change the Way You Work*, Que Publishing
R4: Dan C. Marinescu, *Cloud Computing: Theory and Practice*, Morgan Kaufmann

---

Web Resources

W1: https://aws.amazon.com/big-data/
W2: https://cloud.google.com/bigquery
W3: https://azure.microsoft.com/en-us/products/hdinsight/

W4: https://docs.docker.com
W5: https://www.databricks.com

Course Code: CBD2508	**Course Title:** Big Data Technologies **Type of Course: Program Core Theory**		**L- T-P- C**	3-0-0-3
Version No.	1.0			
Course Pre-requisites				
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.				
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.				

Catalogue prepared by	Dr. Senthilkumar S Ms. Bhoomika A P Mr. Amogh P K
Recommended by the Board of Studies on	BOS NO: 16, BOS held on 25/07/22
Date of Approval by the Academic Council	Academic Council Meeting No.18, Dated 03/08/22

Course Code: **CBD2509**	**Course Title:** Big Data Technologies Laboratory **Type of Course: Program Core** **Theory**	**L- T-P- C**	0-0-4-2
Version No.	1.0		
Course Pre-requisites			
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.				

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.

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 fileand 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:

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.

Catalogue prepared by	Dr. Senthilkumar S Ms. Bhoomika A P Mr. Amogh P K
Recommended by the Board of Studies on	BOS NO: 16, BOS held on 25/07/22
Date of Approval by the Academic Council	Academic Council Meeting No.18, Dated 03/08/22

Course Code: PG COURSE: CBD2510	**Course Title:NoSQL Databases** **Type of Course:Program Core Theory**	**L-T-P-C**	3-0-0-3
Version No.	1.0		
Course Pre-requisites	CSE2074-DBMS		
Anti-requisites	NIL		
Course Description	Introduction to non-relational (NoSQL) data models, such as Key-Value, Document, Column, Graph and Object-Oriented database models. Advantages and disadvantages of the different data architecture patterns will be discussed. Hands-on experience with a representative sample of open-source NoSQL databases will be provided. The rapid and efficient processing of data sets with a focus on performance, reliability, and agility will be covered.		

Course Objectives	The objective of the course is to familiarize the learners with the concepts of **NoSQL Databases** and attain **Skill Development** through **Experiential Learning** techniques.			
Course Out Comes	On successful completion of the course the students shall be able to: 1. **Understand**history, fundamentals,characteristics, and main benefits of NoSQL databases. [Knowledge] 2.**Comprehend**different types of NoSQL databases through case studies. [Comprehension] 3. **Design**different types of NoSQL databases, add content, and try queries on them. [Comprehension]			
Course Content:				
Module 1	NoSQL Database Architectures	Assignment	Knowledge	**No. ofClasses:10**
Topics: Transactions: Concurrency and Integration, ACID, NoSQL emergence and its main features, BASE for reliable database transactions, Achieving horizontal scalability with data base sharding, Brewers CAP theorem. Main Data models of NoSQL: Document Data Model, Key-Value Data Model, Columnar Data Model, Graph Data Model.				
Module 2	Document data model	Assignment	Analysis	**No. of Classes:11**
Topics: Characteristics of Document Data Model, Collection, Naming, CRUD Operation, Querying, Indexing, Replication, Sharding, Consistency, Update Consistency, Read Consistency, Relaxing Consistency, Capped Collection.				
Module 3	Document Data Model Hands on: Mongo DB/Casandra	Assignment	Programming (EmbeddedLab)	**No. ofClasses:9**
Topics:Install, Perform CRUD (create, read, update and delete) Operations, Aggregations, Data Models, Transactions, Indexes, Security, Replication and Sharding.				
Module 4	**Basics of Columnar and Graph Data Models**	Assignment	Comprehend	**No. ofClasses:15**
Topics: Columnar Data Model: Comparison of columnar and row-oriented storage, Column-store Architectures: C-Store and Vector-Wise, Column-store internals and, Inserts/updates/deletes, Indexing, Adaptive Indexing and Database Cracking. Graph Data Model: Comparison of Relational and Graph Modeling, Property Graph Model Graph Analytics: Link analysis algorithm- Web as a graph, Page Rank-Markov chain, page rank computation, Topic specific page rank (Page Ranking Computation techniques: iterative processing, Random walk distribution.				
Learn MongoDB/Casandra by doing the following • Master the art of queries, CRUD, schema design, and data aggregation • Understand scalability using sharding and replication • Write code, build real-world projects and learn hands-on with Cloud Labs				
Project work/Assignment: Mention the Type of Project /Assignment proposed for this course				
Project Works: **1. Create a database that stores road cars. Cars have a manufacturer, a type. Each car has a maximum performance and a maximum torque value. Do the following: Test Cassandras replication schema and Consistency models.** **2. Shopping Mall case study using cassendra, where we have many customers ordering items from the mal land we have suppliers who deliver them their ordered items.**				

Text Books

1. Sadalage, P. & Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Wiley Publications,1st Edition,2019
   https://bigdata-ir.com/wp-content/uploads/2017/04/NoSQL-Distilled.pdf
2. Bradshaw &Chodorow. *MongoDB: The Definitive Guide: Powerful and Scalable Data Storage*, 3rd ed., O'Reilly, 2019
   https://www.oreilly.com/library/view/mongodb-the-definitive/9781491954454/

References

1. Pivert. *NoSQL Data Models: Trends and Challenges*, 1st ed. Wiley, 2018
   https://www.perlego.com/book/995563/nosql-data-models-trends-and-challenges-pdf

2. Amit Phaltankar, Juned Ahsan, Michael Harrison, LiviuNedov, MongoDB Fundamentals A hands-on guide to using MongoDB and Atlas in the real world: 1st edition, Packt publications, 2020
   https://www.perlego.com/book/2059687/mongodb-fundamentals-a-handson-guide-to-using-mongodb-and-atlas-in-the-real-world-pdf

More than 25% of changes are made from the earlier version. Changesare highlighted in bold.

Topics relevant to "SKILL DEVELOPMENT": Usage of un-structured data for Skill Development through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.

Catalogue prepared by	Dr. Naga Raju Mysore, Dr.Senthilkumar
Recommended by the Board of Studies on	BOS NO: 16 th. BOS held on 25/07/22
Date of Approval by the Academic Council	Academic Council Meeting No. 18, Dated 03/08/22

Course Code: COURSE: CBD2511	**Course Title:NoSQL Databases Lab** **Type of Course:Program Core Theory**	**L-T-P-C**	0-0-4-2
Version No.	1.0		
Course Pre-requisites			
Anti-requisites	NIL		
Course Description	Introduction to non-relational (NoSQL) data models, such as Key-Value, Document, Column, Graph and Object-Oriented database models. Advantages and disadvantages of the different data architecture patterns will be discussed. Hands-on experience with a representative sample of open-source NoSQL databases will be provided. The rapid and efficient processing of data sets with a focus on performance, reliability, and agility will be covered.		
Course Objectives	The objective of the course is to familiarize the learners with the concepts of **NoSQL Databases** and attain **Skill Development** through **Experiential Learning** techniques.		
Course Out Comes	On successful completion of the course the students shall be able to: 4. **Understand**history, fundamentals,characteristics, and main benefits of NoSQL databases. [Knowledge] 5.**Comprehend**different types of NoSQL databases through case studies. [Comprehension] 6. **Design**different types of NoSQL databases, add content, and try queries on them. [Comprehension]		
Course Content:			

Module 1	NoSQL Database Architectures	Assignment	Knowledge	**No. of Classes:6**
Topics: Transactions: Concurrency and Integration, ACID, NoSQL emergence and its main features, BASE for reliable database transactions, Achieving horizontal scalability with data base sharding, Brewers CAP theorem. Main Data models of NoSQL: Document Data Model, Key-Value Data Model, Columnar Data Model, Graph Data Model.				
Module 2	Document data model	Assignment	Analysis	**No. of Classes:6**
Topics: Characteristics of Document Data Model, Collection, Naming, CRUD Operation, Querying, Indexing, Replication, Sharding, Consistency, Update Consistency, Read Consistency, Relaxing Consistency, Capped Collection.				

Module 3	Document Data Model Hands on: Mongo DB/Casandra	Assignment	Programming (Embedded Lab)	**No. of Classes:7**
Topics:Install, Perform CRUD (create, read, update and delete) Operations, Aggregations, Data Models, Transactions, Indexes, Security, Replication and Sharding.				
Module 4	**Basics of Columnar and Graph Data Models**	Assignment	Comprehend	**No. of Classes:7**

Topics:
Columnar Data Model: Comparison of columnar and row-oriented storage, Column-store Architectures: C-Store and Vector-Wise, Column-store internals and, Inserts/updates/deletes, Indexing, Adaptive Indexing and Database Cracking.
Graph Data Model: Comparison of Relational and Graph Modeling, Property Graph Model Graph Analytics: Link analysis algorithm- Web as a graph, Page Rank-Markov chain, page rank computation, Topic specific page rank (Page Ranking Computation techniques: iterative processing, Random walk distribution.

- **List of Lab Experiments**

Lab Experiments are to be conducted on the following topics
Topic 1: Install MongoDB
Topic 2: Do lab experiment to perform CRUD (create, read, update and delete).
Topic 2: Demonstrate Aggregations in NoSQL with a real-life application.
Topic 3: Demonstrate different aspect of transactions in NoSQL by taking suitable problem.
Topic 5: Show making indexes in NoSQL with a suitable application.
Topic 6: Illustrate security features of NoSQL with a suitable problem. Topic 6: Explain Sharding concept practically through a suitable example.

Targeted Applications(few are as given below):

1. Content Management systems are pretty common. All the comments on posts on social media are contained in a separate database. In MongoDB, a model has been designed to store such comments and is known as “MetaData and Asset Management”.
2. MongoDB is widely used for storing product information and details by finance and e-commerce companies. You can even store the product catalogue of your brand in it.
3. MongoDB can also be used to store and model machine-generated data. For this, you can learn the “Storing Log data” document. This is known as operational intelligence.

List of MongoDB Tools

- MongoDB Compass.
- Mongo Management Studio.
- MongoJS Query Analyzer.
- Nucleon Database Master.
- NoSQLBooster.
- Studio 3T.
- MongoDB Spark Connector.

3. MongoDB Charts.

Text Books

3. Sadalage, P. & Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Wiley Publications,1st Edition,2019
https://bigdata-ir.com/wp-content/uploads/2017/04/NoSQL-Distilled.pdf
4. Bradshaw &Chodorow. *MongoDB: The Definitive Guide: Powerful and Scalable Data Storage*, 3rd ed., O'Reilly, 2019
https://www.oreilly.com/library/view/mongodb-the-definitive/9781491954454/

References

3. Pivert. *NoSQL Data Models: Trends and Challenges*, 1st ed. Wiley, 2018 https://www.perlego.com/book/995563/nosql-data-models-trends-and-challenges-pdf

4. Amit Phaltankar, Juned Ahsan, Michael Harrison, LiviuNedov, MongoDB Fundamentals A hands-on guide to using MongoDB and Atlas in the real world: 1st edition, Packt publications, 2020 https://www.perlego.com/book/2059687/mongodb-fundamentals-a-handson-guide-to-using-mongodb-and-atlas-in-the-real-world-pdf

More than 25% of changes are made from the earlier version. Changesare highlighted in bold.

Topics relevant to "SKILL DEVELOPMENT": Usage of un-structured data for Skill Development through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: CBD2513	**Course Title: Web Intelligence and Analytics** **Type of Course: Lab**		**L- T-P- C**	2-0-0-2
Version No.	1.0			
Course Pre-requisites	CSE2021-Data Mining			
Anti-requisites				
Course Description	This course is an introduction to Web Analytics and Web Intelligence - is not intended to provide an in-depth review of marketing principles and concepts. Nor is it intended to provide an in depth explanation or review of statistical analysis principles, though some of these principals and concepts will be mentioned from time to time in the lectures and reading materials. Rather, this course will give you the mastery of analytics to a sufficient degree to deploy Web Analytics platforms within your organizations and gain meaningful insights from them that can drive the bottom line.			
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Web Intelligence and Analytics** and attain **Skill Development** through **Experiential Learning** techniques.			
Course Out Comes	**On successful completion of the course the students shall be able to:** 1. A grounded understanding of web intelligence and business analytics terminology related to the above. 2. How to deploy web intelligence to improve the outcomes of your marketing or business plan. 3. How Analysts impact the bottom line (their role) within various businesses and lines of business 4. Growth potentials for Web Analysts and Big Data professionals			
Course Content:				
Module 1	INTRODUCTION TO INTELLIGENT WEB	Assignment	Data Collection/Interpretation	**6Sessions**

INTRODUCTION TO INTELLIGENT WEB -Inside the search engine - Examples of intelligent web applications - Basic elements of intelligent applications - Machine learning, data mining – Searching, Reading, indexing, and searching.				
Module 2	LISTEN AND LOAD	Case studies / Case let	Case studies / Case let	**6 Sessions**
LISTEN AND LOAD- Streams, Information and Language, - Statistics of Text - Analyzing Sentiment and Intent – Load - Databases and their Evolution, Big data Technology and Trends.				
Module 3	CLUSTERING AND CLASSIFICATION	Quiz	Case studies / Case let	**9 Sessions**
CLUSTERING AND CLASSIFICATION An overview of clustering algorithms - Clustering issues in very large datasets - The need for classification - Automatic categorization of emails and spam filtering - Classification with very large datasets - Comparing multiple classifiers on the same data.				
Module4- REASONING (4 hours) Reasoning: Logic and its Limits, Dealing with Uncertainty - MechanicalLogic - The Semantic Web - Limits of Logic - Description and Resolution - Collective Reasoning. Module-5 PREDICTING (6 hours) Statistical Forecasting - Neural Networks - Predictive Analytics - Sparse Memories - Sequence Memory - Network Science – Data Analysis: Regression and Feature Selection - Case Study - set of retrieved and processed news stories.				
List of Laboratory Tasks: Laboratory Work: to analyzing the web for various functionalities given in the subject and using various tools and technologies to do the experimentation. It also involves installation and working on tools and technologies in this domain.				
Text Book 1. Gautam Shroff, “Intelligent Web - Search, Smart Algorithms, and Big Data”, Oxford University Press, 2016. 2. HaralambosMarmanis, Dmitry Babenko, “Algorithms of the Intelligent Web”, Manning publications, 2019.				
References hristopher D. Manning, PrabhakarRaghavan, HinrichSchütze, “An Introduction to Information Retrieval”, Cambridge University Press, 2019. . Mark Gardener, “Beginning R - The Statistical Programming Language”, John Wiley & Sons, Inc., 2012. . W. N. Venables, D. M. Smith and the R Core Team, “An Introduction to R”, 2013. R3 **ɔ resources:** ://www.coursetalk.com/coursera/web-intelligence-and-big-data Course code Course Title L T nformatics.global, **s://sm-nitk.vlabs.ac.in/**				
Topics relevant to “Skill Development”: Intelligent Web and Clustering for Skill Development through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.				
Catalogue prepared by	Dr.Senthilkumar			
Recommended bythe Board of Studies on	BOS NO: 16th BOS held on 25.07.2022			
Date of Approval by the Academic Council	Academic Council meeting no. 18 dated 03.08.2022			
Targeted Application & Tools that can be used				

Course Code: CBD2514	**Course Title: Web Intelligence and Analytics Lab** **Type of Course: Theory**		**L- T-P- C**	2-0-0-2
Version No.	1.0			
Course Pre-requisites	CSE2021-Data Mining			
Anti-requisites				
Course Description	This course is an introduction to Web Analytics and Web Intelligence - is not intended to provide an in-depth review of marketing principles and concepts. Nor is it intended to provide an in depth explanation or review of statistical analysis principles, though some of these principals and concepts will be mentioned from time to time in the lectures and reading materials. Rather, this course will give you the mastery of analytics to a sufficient degree to deploy Web Analytics platforms within your organizations and gain meaningful insights from them that can drive the bottom line.			
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Web Intelligence and Analytics** and attain **Skill Development** through **Experiential Learning** techniques.			
Course Out Comes	**On successful completion of the course the students shall be able to:** A grounded understanding of web intelligence and business analytics terminologyrelated to the above. How to deploy web intelligence to improve the outcomes of your marketing or business plan. How Analysts impact the bottom line (their role) within various businesses and lines of business Growth potentials for Web Analysts and Big Data professionals			
Course Content:				
Module 1	INTRODUCTION TO INTELLIGENT WEB	Assignment	Data Collection/Interpretation	**6Sessions**
INTRODUCTION TO INTELLIGENT WEB -Inside the search engine - Examples of intelligent web applications - Basic elements of intelligent applications - Machine learning, data mining – Searching, Reading, indexing, and searching.				
Module 2	LISTEN AND LOAD	Case studies / Case let	Case studies / Case let	**6 Sessions**
LISTEN AND LOAD- Streams, Information and Language, - Statistics of Text - Analyzing Sentiment and Intent – Load - Databases and their Evolution, Big data Technology and Trends.				
Module 3	CLUSTERING AND CLASSIFICATION	Quiz	Case studies / Case let	**9 Sessions**
CLUSTERING AND CLASSIFICATION An overview of clustering algorithms - Clustering issues in very large datasets - The need for classification - Automatic categorization of emails and spam filtering - Classification with very large datasets - Comparing multiple classifiers on the same data.				
Module4- REASONING (4 hours) Reasoning: Logic and its Limits, Dealing with Uncertainty - MechanicalLogic - The Semantic Web - Limits of Logic - Description and Resolution - Collective Reasoning. Module-5 PREDICTING (6 hours) Statistical Forecasting - Neural Networks - Predictive Analytics - Sparse Memories - Sequence Memory - Network Science – Data Analysis: Regression and Feature Selection - Case Study - set of retrieved and processed news stories.				
List of Laboratory Tasks: Laboratory Work: to analyzing the web for various functionalities given in the subject and using various tools and technologies to do the experimentation. It also involves installation and working on tools and technologies in this domain.				

Week	Lab Experiment Title	Tools / Technologies
1	Design a static web page using HTML and CSS	HTML5, CSS3
2	Create an interactive web form with validation using JavaScript	JavaScript, DOM
3	Build a dynamic dashboard with charts using any JS library (e.g., Chart.js / Google Charts)	JavaScript, Chart.js / Google Charts
4	Introduction to WEKA: Loading datasets and understanding file formats	WEKA (ARFF, CSV)
5	Perform data preprocessing: normalization, filtering, and attribute selection	WEKA Preprocess Panel
6	Apply classification using J48 (Decision Tree) and evaluate performance	WEKA – J48
7	Classification using Naïve Bayes and compare accuracy	WEKA – Naïve Bayes
8	Implement k-Nearest Neighbors (IBk) and visualize confusion matrix	WEKA – IBk
9	Perform clustering using k-Means and interpret results	WEKA – SimpleKMeans
10	Explore hierarchical clustering on a sample dataset	WEKA – HierarchicalClusterer
11	Apply association rule mining using Apriori algorithm	WEKA – Apriori
12	Visualize the rules and patterns discovered from transactional data	WEKA Visualization tools
13	Use WEKA Experimenter to compare classifiers on multiple datasets	WEKA – Experimenter
14	Feature selection using attribute evaluators and rankers	WEKA – Attribute Selection
15	Mini project: Perform end-to-end analysis on a real dataset using classification and visualization	WEKA + any visualization tool (optional)

Text Book

3. Gautam Shroff, “Intelligent Web - Search, Smart Algorithms, and Big Data”, Oxford University Press, 2016.
4. HaralambosMarmanis, Dmitry Babenko, “Algorithms of the Intelligent Web”, Manning publications, 2019.

References

hristopher D. Manning, PrabhakarRaghavan, HinrichSchütze, “An Introduction to Information Retrieval”, Cambridge University Press, 2019.

. Mark Gardener, “Beginning R - The Statistical Programming Language”, John Wiley & Sons, Inc., 2012.

. W. N. Venables, D. M. Smith and the R Core Team, “An Introduction to R”, 2013. R3

o resources:

://www.coursetalk.com/coursera/web-intelligence-and-big-data Course code Course Title L T

nformatics.global,

s://sm-nitk.vlabs.ac.in/

Topics relevant to “Skill Development”: Intelligent Web and Clustering for Skill Development through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.

Catalogue prepared by	Dr.Senthilkumar
Recommended bythe Board of Studies on	BOS NO: 16th BOS held on 25.07.2022
Date of Approval by the Academic Council	Academic Council meeting no. 18 dated 03.08.2022

Targeted Application & Tools that can be used

Course Code: CSE1700	**Course Title:** Essentials of AI **Type of Course:** Theory	**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:

1. **Data Preprocessing**: Clean and manipulate data from various sources such as CSV, Excel, SQL databases, and APIs.
2. **Exploratory Data Analysis (EDA)**: Gain insights into datasets by identifying trends, patterns, and outliers.
3. **Predictive Modeling**: Build models for classification (e.g., spam detection) and regression (e.g., house price prediction).
4. **Clustering**: Group data into clusters for unsupervised learning tasks (e.g., customer segmentation).
5. **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: 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: 1. Proficiency in Implementing AI Algorithms Using Python 2. Ability to Build and Evaluate Machine Learning Models 3. Hands-on Experience with Neural Networks and Deep Learning 4. Practical Application of AI to Solve Real-World Problems
Course Content:	

Module 1	Introduction to AI and Python for AI	ssignment	mplementation	**8 Sessions**

Lab Assignment 1: Setting Up the Python Environment

- **Objective:** Get familiar with setting up a Python environment for AI projects.
- **Tasks:**
    1. Install Python, Anaconda, and Jupyter Notebook.
    2. Set up a virtual environment for AI development.
    3. Install essential Python libraries: numpy, pandas, matplotlib, and scikit-learn.
    4. 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:**
    1. Write Python code to work with basic data types (integer, float, string, boolean).
    2. Implement and manipulate Python lists, tuples, sets, and dictionaries.
    3. Create basic control flow structures: if-else, for loops, while loops.
    4. 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:**
    1. Load a dataset (e.g., Titanic or Iris dataset) using pandas.
    2. Clean the dataset by handling missing values, removing duplicates, and

converting data types if needed.

3. Explore the dataset by visualizing it using matplotlib and seaborn.
4. 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	8 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:

1. **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().
2. **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().
3. **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.
4. **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:

1. **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.

2. **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.
3. **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).
4. **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:

1. **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.
2. **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.
3. **Distribution Visualizations:**
    - Plot distributions of continuous variables using **Seaborn's** distplot() or kdeplot().
    - Create bar plots for categorical variables to understand their frequency distribution.
4. **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:

1. **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.
2. **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.
3. **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.
4. **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:

1. **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().
2. **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.
3. **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.

4. **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	8 Sessions

Lab Assignment 3: Implementing Linear Regression

- **Tasks:**
    1. Load a real-world dataset (e.g., **Boston Housing Price** dataset).
    2. Train a **Linear Regression** model using LinearRegression() from scikit-learn.
    3. Evaluate the model using **Mean Squared Error (MSE)** and **R-squared Score**.
    4. Visualize the regression line using Matplotlib.

Lab Assignment 4: Logistic Regression for Classification

- **Tasks:**
    1. Load the **Iris** or **Breast Cancer** dataset.
    2. Preprocess the dataset (handle missing values, encode categorical variables, scale data).
    3. Train a **Logistic Regression** model using LogisticRegression().
    4. Evaluate performance using **Accuracy, Precision, Recall, F1-score**.
    5. Plot the **Confusion Matrix** and **ROC Curve**.

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

- **Tasks:**
    1. Load the **Iris dataset** and split it into training and testing sets.
    2. Train a **KNN classifier** using KNeighborsClassifier().
    3. Experiment with different values of **K** and evaluate performance.
    4. Visualize decision boundaries using a **scatter plot**.

Lab Assignment 6: Decision Trees and Random Forests

- **Tasks:**
    1. Train a **Decision Tree classifier** on the Titanic dataset.
    2. Visualize the tree structure using plot_tree().
    3. Train a **Random Forest classifier** and compare performance with the decision tree.
    4. Determine the **feature importance** using feature_importances_.

Module 4	Neural Networks and Deep Learning	Quiz	Implementation	6 Sessions

Lab Assignment 7: Introduction to Perceptron and Activation Functions

Tasks:

1. Implement a **single-layer perceptron** using NumPy.
2. Train the perceptron to classify **AND, OR, XOR** gates.
3. Experiment with different **activation functions** (Sigmoid, ReLU, Tanh).
4. Visualize decision boundaries.

Lab Assignment 8: Building a Simple Neural Network with Keras

Tasks:

1. Load the **MNIST dataset** from keras.datasets.
2. Preprocess the data (normalize pixel values, reshape input).
3. Create a **fully connected neural network** using Sequential API.
4. Train and evaluate the model using **categorical cross-entropy loss** and **accuracy**.

Lab Assignment 9: Implementing CNN from Scratch

Tasks:

1. Load the **CIFAR-10 dataset**.
2. Build a CNN with **Conv2D, MaxPooling2D, Flatten, Dense, Dropout** layers.
3. Use **Adam optimizer** and **categorical cross-entropy loss**.
4. Train and visualize loss/accuracy curves.

Lab Assignment 10: Image Augmentation & Regularization

Tasks:

1. Apply **data augmentation** (rotation, zoom, flipping) using ImageDataGenerator.
2. Add **dropout and batch normalization** to prevent overfitting.
3. Compare model performance with and without augmentation.

Lab Assignment 11: Transfer Learning with Pre-trained Models

Tasks:

1. Use **VGG16 or ResNet50** pre-trained on ImageNet.
2. Replace the output layer to classify **new images**.
3. Freeze earlier layers and fine-tune deeper layers.

4. Evaluate the model on a custom **dataset (e.g., Cats vs. Dogs)**.

Lab Assignment 12: Implementing RNN for Text Classification

Tasks:

1. Load **IMDB movie reviews dataset** from keras.datasets.
2. Preprocess text (tokenization, padding sequences).
3. Build an **RNN** with **Embedding, SimpleRNN, Dense** layers.
4. Train and evaluate the model.

Lab Assignment 13: Building an LSTM for Time Series Prediction

Tasks:

1. Load a **time series dataset** (e.g., stock prices, temperature data).
2. Preprocess the data (normalize, reshape).
3. Build an **LSTM-based model**.
4. Predict future values and visualize trends.

Targeted Application & Tools that can be used:

Applications:

1. **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):

1. "Artificial Intelligence with Python" – Prateek Joshi
2. "Python Machine Learning" – Sebastian Raschka & Vahid Mirjalili
3. "Hands-On Artificial Intelligence with Python" – Teet Straus
4. "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/

Catalogue prepared by	
Recommended by the Board of Studies on	
Date of Approval by the Academic Council	

Course Code: **CSE2509**	**Course Title:** Mobile Applications and Development Lab **Type of Course:** Lab	**L- T-P- C**	0	**0**	4	**2**
Version No.	2.0					
Course Pre-requisites	CSE1514 Object Oriented Programming using Java					
Anti-requisites	**NIL**					
Course Description	The course provides hands-on experience in designing, developing, and deploying mobile applications for Android and iOS platforms. Students will work with native development frameworks such as Android Studio (Java/Kotlin) and Xcode (Swift), as well as explore cross-platform tools like Flutter or React Native.					
Course Objective	The objective of the course is to develop Native and Cross-Platform Mobile Applications, design Interactive and Responsive User Interfaces, integrate Backend Services and APIs, implement State Management and Performance Optimization, ensure Mobile App Security and Data Protection					
Course Outcomes	On successful completion of the course the students shall be able to: 1. Develop Functional Mobile Applications 2. Design and Implement Interactive UIs 3. Integrate Cloud Services and APIs 4. Integrate Backend Systems and Data Management					

	5. Deploy, Publish, and Maintain advanced Mobile Application			
Course Content:				
Module 1	Introduction and Architecture of Android	Assignment	Simulation/Data Analysis	8 Sessions
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. Design an app to input your personal information. Use an autocomplete text view to select your place of birth.				
Module 2	User Interfaces, Intent and Fragments	Term paper/Assignment	Simulation/Data Analysis	13 Sessions
3. a. 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. b. Design a restaurant menu app to print the total amount of orders.				
Module 3	Components of Android	Term paper/Assignment	Simulation/Data Analysis	13 Sessions
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.				
Module 4	Notifications and Data Persistence	Term paper/Assignment	Simulation/Data Analysis	13 Sessions
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.

Module 5	Advance App Development	Term paper/Assignment	Simulation/Data Analysis	13 Sessions

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:

Applications:

1. Native Android Applications (Java/Kotlin)
   - Android Mobile Apps built for Android smartphones and tablets using Java or Kotlin programming languages.
   - Target audience: Android users.
2. Native iOS Applications (Swift)
   - iOS Mobile Apps designed for iPhone and iPad using Swift.
   - Target audience: iOS users (Apple ecosystem).
3. Cross-Platform Mobile Apps (Flutter, React Native)
   - Cross-platform apps designed to run on both Android and iOS from a single codebase using frameworks like Flutter or React Native.

- Target audience: Users on both Android and iOS platforms.

4. Mobile Web Applications (Progressive Web Apps - PWA)

- Mobile-optimized web applications using HTML5, CSS3, and JavaScript that run in a browser with native-like functionality (offline support, push notifications).
- Target audience: Users accessing apps via mobile browsers.

Development Tools and Frameworks

1. Integrated Development Environments (IDEs)

- Android Studio (for Android): The official IDE for Android development, supporting Java, Kotlin, and Android SDK.
- Xcode (for iOS): The official IDE for iOS development with Swift and Objective-C, providing a comprehensive suite of development tools for iPhone/iPad applications.
- Visual Studio Code (VS Code): Lightweight IDE for working with Flutter, React Native, and web development projects.

2. Cross-Platform Development Frameworks

- Flutter: Open-source UI framework by Google for building natively compiled applications for mobile, web, and desktop from a single codebase.
- React Native: Open-source framework developed by Facebook for building cross-platform apps with JavaScript and React.

3. Backend & Cloud Tools

- Firebase: Google's backend-as-a-service (BaaS) platform offering authentication, real-time databases, cloud storage, and push notifications for mobile apps.
- AWS Amplify: Cloud platform for backend services (API, storage, authentication) and mobile deployment.
- SQLite / Realm: Local storage solutions for mobile apps to manage data storage and retrieval on-device.

4. Mobile App Testing and Debugging Tools

- Android Emulator (for Android): A virtual device to run and test Android apps without needing physical devices.
- Xcode Simulator (for iOS): A tool to simulate different iOS devices and test apps during development.
- Appium: Open-source tool for automated testing across native, hybrid, and mobile web applications.

5. Version Control and Collaboration

- Git: Version control system for managing code changes and collaborating with teams.

- GitHub / GitLab / Bitbucket: Online platforms for hosting Git repositories, collaboration, and version control management.

6. Mobile App Deployment Tools

- Google Play Console: For managing Android app publishing, distribution, and monitoring.
- Apple App Store Connect: For managing iOS app submissions, reviews, and releases on the Apple App Store.

7. UI/UX Design Tools

- Figma / Adobe XD: Tools for UI/UX design and wireframing to create the visual elements of mobile applications before development.
- Sketch: Vector-based design tool for iOS UI design and prototyping

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/

Catalogue prepared by	
Recommended by the Board of Studies on	
Date of Approval by the Academic Council	

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.					
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)

Course Code: CBD2515	**Bioinformatics and Genomic Data Analytics**	**L-T-P-C**	3	0	0	3
Version No.	1.0					
Course Pre-requisites	CBD 1700 Introduction to Big Data					
Anti-requisites	NIL					
Course Description	This course introduces the fundamental concepts and tools of bioinformatics and genomic data analysis. It covers biological databases, sequence alignment, gene expression analysis, and genome annotation. Students will learn computational techniques for analyzing large-scale genomic and transcriptomic datasets. The course emphasizes the application of statistical, algorithmic, and machine learning approaches in biological data interpretation. Practical sessions include hands-on experience with tools such as BLAST, Bioconductor, and R for genomic data analysis.					
Course Objectives	To provide foundational knowledge of molecular biology concepts essential for understanding genomic data and bioinformatics tools. To introduce key algorithms and statistical methods used in sequence alignment, gene prediction, and genome annotation. To equip students with practical skills in analyzing and interpreting high-throughput genomic and transcriptomic data using open-source bioinformatics tools. To enable critical thinking and application of computational approaches for solving biological research problems and data-driven discoveries in genomics.					
Course Out Comes	CO1 : Describe the fundamental concepts of bioinformatics, including sequence alignment, gene annotation, and structural genomics.(Understand) CO2 :Analyse genomic and proteomic data to interpret biological significance using statistical and computational tools.(Analyse) CO3: Apply bioinformatics tools and algorithms for sequence comparison, genome annotation, and phylogenetic analysis. (Apply) CO4: Apply data analytics techniques to process and visualize large-scale					

	genomic data using R, Python, or similar platforms. (Apply)			
Module 1	**Introduction to Bioinformatics and Genomics**		**Understand**	**No. of Sessions: 10**
Overview of Bioinformatics and its Applications, Central Dogma of Molecular Biology, DNA, RNA, and Protein Structures, Introduction to Genomics and Proteomics, Biological Databases: NCBI, EMBL, UniProt, File Formats: FASTA, GenBank				
Module 2	**Sequence Alignment and Genome Annotation**		**Analyse**	**No. of Sessions: 12**
Pairwise and Multiple Sequence Alignment, Dynamic Programming: Needleman-Wunsch and Smith-Waterman Algorithms, BLAST and FASTA Tools, Gene Prediction and Annotation Techniques, Homology Modelling, Comparative Genomics, Case Studies on Gene Function Prediction				
Module 3	**Phylogenetics and Structural Bioinformatics**		**Apply**	**No. of Sessions: 11**
Phylogenetic Tree Construction: UPGMA, Neighbor Joining, Molecular Evolution and Substitution Models, Protein Structure Prediction, Protein-Protein Interaction Networks, Tools: MEGA, Clustal Omega, SWISS-MODEL				
Module 4	**Genomic Data Analytics**		**Apply**	**No. of Sessions: 12**
Next-Generation Sequencing (NGS) Technologies, Data Preprocessing and Quality Control, Gene Expression Data Analysis (Microarray, RNA-Seq), Tools and Languages: Bioconductor (R), Pandas, BioPython, Data Visualization Techniques, Clustering Techniques, Introduction to Machine Learning in Genomics				

Textbooks:

T1: Arthur M. Lesk, *Introduction to Bioinformatics*, Oxford University Press, 5th Edition, 2019.

T2: Jonathan Pevsner, *Bioinformatics and Functional Genomics*, Wiley-Blackwell, 3rd Edition, 2015

Reference Books:

R1: Neil C. Jones and Pavel A. Pevzner, *An Introduction to Bioinformatics Algorithms*, MIT Press, 2004.

R2: R.M. Lathe, *Genomics: The Science and Technology Behind the Human Genome Project*, Springer, 2004.

R3: Andreas D. Baxevanis, B. F. Francis Ouellette, *Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins*, Wiley-Interscience, 3rd Edition, 2004.

R4: Richard Durbin, Sean R. Eddy, Anders Krogh, Graeme Mitchison, *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*, Cambridge University Press, 1998.

Web Resources:

W1: NCBI Bioinformatics Tools – https://www.ncbi.nlm.nih.gov/tools/

W2: EMBL-EBI Training Resources – https://www.ebi.ac.uk/training/

Course Code: CBD2516	**Bigdata in Supply Chain Logistics**	**L-T-P-C**	3	0	0	3
Version No.	1.0					
Course Pre-requisites	CBD 1700 Introduction to Big Data					
Anti-requisites	NIL					
Course Description	This course introduces the application of big data analytics in the field of supply chain and logistics management. It explores how data-driven decision-making can enhance supply chain visibility, forecasting accuracy, and operational efficiency. Learners will gain insights into data acquisition, integration, and processing methods using big data tools and platforms tailored to supply chain contexts. Topics include real-time tracking, demand prediction, inventory optimization, risk assessment, and strategic sourcing using big data. The course emphasizes both theoretical understanding and practical implementations using case studies and industry examples.					
Course Objectives	To introduce the concepts and importance of big data analytics in modern supply chain and logistics management. To familiarize students with data sources, data types, and technologies used in supply chain data analytics. To develop analytical skills for interpreting large datasets and extracting meaningful insights for supply chain decisions. To enable learners to apply big data tools and techniques for solving real-time logistics and supply chain problems.					
Course Out Comes	CO1: Understand the fundamentals of big data and its role in enhancing supply chain and logistics operations. CO2: Analyze various big data tools and technologies used for optimizing supply chain processes. CO3: Apply data-driven techniques to improve forecasting, inventory control, and demand planning. CO4: Apply big data analytics to solve real-world supply chain and logistics					

	challenges, ensuring better decision-making and efficiency.			
Module 1	**Introduction to Big Data in Supply Chain**		**Understand**	**No. of Sessions: 10**
Introduction to supply chain and logistics, Basics of Big Data, Role of data in supply chain, Data generation sources (IoT, RFID, GPS), Data types and challenges in supply chain analytics, Use cases and industry relevance.				
Module 2	**Big Data Technologies and Tools for Supply Chain**		**Analyse**	**No. of Sessions: 12**
Big Data ecosystem (Hadoop, Spark), Data storage frameworks (HDFS, NoSQL), Real-time vs. batch processing, Data integration platforms, Overview of data warehousing and ETL tools, Role of cloud computing in supply chain logistics.				
Module 3	**Data Analytics and Predictive Modelling**		**Apply**	**No. of Sessions: 11**
Descriptive, Predictive, and Prescriptive Analytics, Demand forecasting, Inventory optimization, Transportation analytics, Predictive maintenance, Machine learning models in logistics, Case studies with sample datasets.				
Module 4	**Applications and Case Studies in Supply Chain Analytics**		**Apply**	**No. of Sessions: 12**
Big Data applications in procurement, warehousing, and last-mile delivery, Risk analysis, Supplier performance evaluation, Real-world case studies (Amazon, Walmart, Maersk), Ethical and privacy concerns in data handling.				

Textbooks

T1: Nada R. Sanders, Big Data Driven Supply Chain Management: A Framework for Implementing Analytics and Turning Information into Intelligence, Pearson Education, 2014.

T2: David Stephenson, Big Data Demystified: How to Use Big Data, Data Science and AI to Make Better Business Decisions and Gain Competitive Advantage, Pearson FT Press, 2018.

Reference Books

R1: Michael H. Hugos, *Essentials of Supply Chain Management*, Wiley, 4th Edition, 2018.

R2: Viktor Mayer-Schönberger & Kenneth Cukier, *Big Data: A Revolution That Will Transform How We Live, Work, and Think*, Eamon Dolan/Houghton Mifflin Harcourt, 2013.

R3: Arvind Sathi, *Big Data Analytics: Disruptive Technologies for Changing the Game*, IBM Press, 2012.

R4: Thomas H. Davenport & Jeanne G. Harris, *Competing on Analytics: The New Science of Winning*, Harvard Business Review Press, 2017.

Web Resources

W1: https://www.supplychaindigital.com/ – News and trends in supply chain and logistics.

W2: https://www.scmr.com/ – Supply Chain Management Review articles and case studies.

W3: https://dataflog.com/ – Big data news, trends, and insights across industries.

W4: https://www.ibm.com/topics/supply-chain-analytics – IBM's perspective on analytics in supply chain.
W5: https://hbr.org/ – Harvard Business Review articles on business analytics and data-driven decision making.

Course Code: CBD2517	**Bigdata in Supply Chain Logistics Lab**	**L-T-P-C**	0	0	2	1
Version No.	1.0					
Course Pre-requisites	CBD 1700 Introduction to Big Data					
Anti-requisites	NIL					
Course Description	This lab course is designed to provide hands-on experience in applying big data tools and analytics to real-world supply chain and logistics scenarios. Students will work with datasets involving inventory, transportation, warehousing, and demand forecasting to extract insights and optimize logistics operations. Using tools such as Hadoop, Spark, Python, and Tableau, students will explore data collection, preprocessing, visualization, and predictive analytics techniques to improve decision-making across supply chain networks.					
Course Objectives	To impart practical knowledge on the application of big data tools in the context of supply chain logistics. To enable students to perform data preprocessing, cleaning, and transformation using relevant datasets. To analyze and visualize logistics data using big data analytics platforms for better decision-making. To develop predictive models for demand forecasting, inventory optimization, and transportation planning.					
Course Out Comes	CO1: Understand the role and application of big data analytics in supply chain logistics through hands-on practice. CO2: Analyze and preprocess large supply chain datasets using tools like Hadoop, Spark, and Python. CO3: Apply data visualization techniques to interpret logistics data for strategic planning. CO4: Develop and evaluate predictive models to solve real-world logistics problems.					

List of Tools:

- Apache Hadoop-Distributed storage and processing framework for large datasets.
- Apache Spark-Real-time data processing and analytics engine.
- Apache Flink-Stream and batch data processing tool ideal for real-time logistics data.
- Jupyter Notebook- Ideal for data exploration, visualization, and machine learning using Python.
- R Studio (with tidyverse, dplyr)- Statistical computing and visualization for supply chain data analysis.

List of Experiments:

1 Introduction to Big Data and Hadoop Ecosystem – Setup HDFS and run basic HDFS commands

2 Ingesting supply chain data into HDFS using Apache Flume and Sqoop

3 Analyzing logistics datasets using Apache Hive – Creating tables and querying

4 Data cleaning and transformation using Apache Pig

5 Batch processing of shipment data using Apache Spark (PySpark)

6 Stream processing of warehouse inventory using Apache Flink

7 Predictive analytics on delivery time using machine learning in Jupyter Notebook

8 Real-time logistics data visualization using Elasticsearch and Kibana

9 Design of data pipelines for order tracking using Talend Open Studio

10 Route optimization and geospatial data handling using Spark with GeoJSON

11 Analyzing transportation costs using KNIME with supply chain KPIs

12 Demand forecasting using time series analysis in Python (Pandas + Prophet)

13 Inventory clustering and segmentation using k-means in Orange or R

14 Dashboard creation for logistics operations using Tableau Public or QlikView

15 Mini Project: End-to-end analysis of a supply chain use-case (student groups)

Textbooks

T1: Nada R. Sanders, Big Data Driven Supply Chain Management: A Framework for Implementing Analytics and Turning Information into Intelligence, Pearson Education, 2014.

T2: David Stephenson, Big Data Demystified: How to Use Big Data, Data Science and AI to Make Better Business Decisions and Gain Competitive Advantage, Pearson FT Press, 2018.

Reference Books

R1: Michael H. Hugos, *Essentials of Supply Chain Management*, Wiley, 4th Edition, 2018.
R2: Viktor Mayer-Schönberger & Kenneth Cukier, *Big Data: A Revolution That Will Transform How We Live, Work, and Think*, Eamon Dolan/Houghton Mifflin Harcourt, 2013.
R3: Arvind Sathi, *Big Data Analytics: Disruptive Technologies for Changing the Game*, IBM Press, 2012.
R4: Thomas H. Davenport & Jeanne G. Harris, *Competing on Analytics: The New Science of Winning*, Harvard Business Review Press, 2017.

Web Resources

W1: https://www.supplychaindigital.com/ – News and trends in supply chain and logistics.
W2: https://www.scmr.com/ – Supply Chain Management Review articles and case studies.
W3: https://datafloq.com/ – Big data news, trends, and insights across industries.
W4: https://www.ibm.com/topics/supply-chain-analytics – IBM's perspective on analytics in supply chain.
W5: https://hbr.org/ – Harvard Business Review articles on business analytics and data-driven decision making.

Course Code: CBD2518	**Data Security and Cryptography**	**L-T-P-C**	3	0	0	3
Version No.	1.0					
Course Pre-requisites	CBD 1700 Introduction to Big Data					
Anti-requisites	NIL					
Course Description	This course offers a comprehensive introduction to the principles and practices of data security and cryptography. It covers classical and modern cryptographic algorithms, key management techniques, authentication protocols, data integrity, and network security mechanisms. Students will gain practical understanding of how cryptographic techniques are used to secure systems and communications, ensuring confidentiality, integrity, and availability of data in real-world applications.					
Course Objectives	To introduce the fundamentals of cryptographic techniques and their mathematical foundations. To understand symmetric and asymmetric encryption algorithms and their real-world applications. To explore cryptographic protocols for authentication, digital signatures, and secure communication.					

	To analyze and evaluate security threats and countermeasures in digital systems and networks.
Course Out Comes	CO1 (Understand): Explain fundamental concepts of cryptography and data security. CO2 (Analyze): Compare and analyze various encryption techniques and cryptographic protocols. CO3 (Apply): Implement encryption/decryption algorithms and simulate secure data communication. CO4 (Apply): Apply knowledge of security mechanisms to assess and solve real- time data security issues.

Module 1	**Introduction to Cryptography and Number Theory**		**Understand**	**No. of Sessions: 12**
Cryptography Basics, Security Attacks, Services, Mechanisms, Symmetric vs. Asymmetric Cryptography, Modular Arithmetic, Euler's Theorem, Fermat's Theorem, Euclidean Algorithm.				
Module 2	**Symmetric and Asymmetric Encryption Algorithms**		**Analyse**	**No. of Sessions: 11**
DES, Triple DES, AES, Blowfish, RC4, RSA Algorithm, ElGamal, Key Distribution and Management, Diffie-Hellman Key Exchange.				
Module 3	**Authentication, Hashing and Digital Signatures**		**Apply**	**No. of Sessions: 11**
Authentication Protocols, Message Authentication Codes (MAC), Hash Functions (SHA, MD5), Digital Signatures, Public Key Infrastructure (PKI), Certificate Authorities.				
Module 4	**Data Security and Network Protocols**		**Apply**	**No. of Sessions: 11**
IPSec, SSL/TLS, Secure Email (PGP, S/MIME), Secure E-commerce, Firewalls, Intrusion Detection Systems (IDS), Blockchain Basics for Security.				

Textbooks

T1: William Stallings, Cryptography and Network Security: Principles and Practice, Pearson, 7th Edition, 2017.

T2: Behrouz A. Forouzan, Cryptography and Network Security, McGraw-Hill Education, 2nd Edition, 2011.

Reference Books

R1: Bruce Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, Wiley, 2nd Edition, 1996.

R2: Douglas R. Stinson, *Cryptography: Theory and Practice*, CRC Press, 4th Edition, 2018.

R3: Charlie Kaufman, Radia Perlman, and Mike Speciner, *Network Security: Private Communication in a Public World*, Prentice Hall, 2nd Edition, 2002.

R4: Christof Paar, Jan Pelzl, *Understanding Cryptography: A Textbook for Students and Practitioners*, Springer, 2010.

Web Resources

W1: https://cryptography.io/ – Python cryptographic library and resources.
W2: https://nvlpubs.nist.gov/ – NIST publications on cryptographic standards.
W3: https://www.owasp.org/ – Open Web Application Security Project for best practices.
W4: https://www.tutorialspoint.com/cryptography/index.htm – Cryptography basics and tutorials.
W5: https://www.coursera.org/learn/crypto – Stanford's free online cryptography course on Coursera.

Course Code: CBD2519	**Data Security and Cryptography Lab**	**L-T-P-C**	0	0	4	2
Version No.	1.0					
Course Pre-requisites	CBD 1700 Introduction to Big Data					
Anti-requisites	NIL					
Course Description	This lab course provides practical exposure to the implementation of various cryptographic algorithms and data security mechanisms. Students will explore symmetric and asymmetric encryption methods, hashing algorithms, digital signatures, and secure data transmission protocols. Through hands-on experiments using programming tools and libraries, the course aims to build a strong foundation in securing digital information and understanding the real-world application of cryptographic techniques.					
Course Objectives	To provide practical understanding of classical and modern cryptographic algorithms. To develop skills in implementing data confidentiality, integrity, and authentication mechanisms. To enable students to simulate secure communication using encryption and hashing techniques. To encourage application of cryptography in real-world security scenarios such as digital signatures, SSL, and secure file transfer.					
Course Out Comes	CO1 (Understand): Demonstrate understanding of fundamental cryptographic principles and data protection techniques. CO2 (Apply): Implement symmetric and asymmetric encryption algorithms using programming languages. CO3 (Apply): Apply hashing and digital signature mechanisms to ensure message integrity and authentication. CO4 (Analyze): Analyze the effectiveness of various cryptographic methods in securing data transmission and storage.					

List of Tools:

- OpenSSL- Command-line tool for implementing SSL/TLS, certificate creation, and encryption algorithms.

- CrypTool- Educational tool to demonstrate and analyze cryptographic algorithms visually.
- Wireshark- Packet analyzer for network protocol analysis and examining secure communication.

Python with Crypto Libraries:

- `PyCryptodome` – modern cryptographic library in Python.
- `cryptography` – widely used high-level cryptographic package.
- `hashlib` – built-in module for hashing (SHA, MD5).

List of Experiments

Week 1: Introduction to Cryptographic Tools
- **Familiarization with CrypTool, OpenSSL, and GnuPG.**
- **Basic encryption/decryption using Caesar Cipher and Monoalphabetic Cipher.**

Week 2: Substitution and Transposition Techniques
- **Implement Playfair Cipher and Hill Cipher.**
- **Perform encryption/decryption and analyze security.**

Week 3: Symmetric Key Cryptography – DES
- **DES algorithm implementation using OpenSSL or Python (PyCryptodome).**
- **File encryption/decryption using DES.**

Week 4: Symmetric Key Cryptography – AES
- **Implement AES encryption in ECB and CBC modes.**
- **Analyze block cipher properties.**

Week 5: Asymmetric Cryptography – RSA
- **Key generation, encryption, and decryption using RSA (Python/Java).**
- **Encrypt messages and files.**

Week 6: Diffie-Hellman Key Exchange Protocol
- **Simulate secure key exchange between two users.**
- **Visualize key agreement and its mathematical basis.**

Week 7: Message Digest and Hashing
- **Use hashing algorithms (MD5, SHA-1, SHA-256) in Python.**
- **Verify message integrity using hash functions.**

Week 8: Digital Signature Implementation
- **Generate and verify digital signatures using GnuPG or Python.**
- **Understand certificate generation and validation.**

Week 9: SSL/TLS Secure Communication
- **Use OpenSSL to create a self-signed certificate.**
- **Setup a basic HTTPS server with TLS security.**

Week 10: Steganography and Cryptanalysis
- **Perform basic steganography (text/image) using open tools.**
- **Introduction to cryptanalysis techniques like brute force, frequency analysis.**

Week 11: Wireless Security and WPA2 Cracking Demo (Ethical)
- **Use Kali Linux tools like Aircrack-ng (in a simulated lab).**

- **Understand vulnerabilities in wireless protocols.**

Week 12: Password Cracking Techniques
- **Perform password hashing and cracking using John the Ripper or Hashcat.**
- **Practice using rainbow tables.**

Week 13: Secure Email Communication Using GPG
- **Encrypt and sign emails with GnuPG.**
- **Simulate secure PGP-based communication.**

Week 14: Network Packet Sniffing and Analysis
- **Use Wireshark to capture and analyze SSL/TLS and encrypted packets.**
- **Identify handshake, certificates, and cipher suites.**

Week 15: Mini Project & Viva
- **Students present a mini project (e.g., secure chat, encrypted file sharing).**
- **Viva voce to evaluate understanding of tools and cryptographic principles.**

Textbooks

T1: William Stallings, Cryptography and Network Security: Principles and Practice, Pearson, 7th Edition, 2017.

T2: Behrouz A. Forouzan, Cryptography and Network Security, McGraw-Hill Education, 2nd Edition, 2011.

Reference Books

R1: Bruce Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, Wiley, 2nd Edition, 1996.

R2: Douglas R. Stinson, *Cryptography: Theory and Practice*, CRC Press, 4th Edition, 2018.

R3: Charlie Kaufman, Radia Perlman, and Mike Speciner, *Network Security: Private Communication in a Public World*, Prentice Hall, 2nd Edition, 2002.

R4: Christof Paar, Jan Pelzl, *Understanding Cryptography: A Textbook for Students and Practitioners*, Springer, 2010.

Web Resources

W1: https://cryptography.io/ – Python cryptographic library and resources.

W2: https://nvlpubs.nist.gov/ – NIST publications on cryptographic standards.

W3: https://www.owasp.org/ – Open Web Application Security Project for best practices.

W4: https://www.tutorialspoint.com/cryptography/index.htm – Cryptography basics and tutorials.

W5: https://www.coursera.org/learn/crypto – Stanford's free online cryptography course on Coursera.

Course Code: CSE2510	**Course Title:** Competitive Programming and Problem Solving **Type of Course: Program Core**	L-T-P-C	0	0		
Version No.	1.0					

Course Pre-requisites	NIL
Anti-requisites	**NIL**
Course 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.
Course 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.
Course 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.

Module 1: Introduction to Competitive Programming

Overview 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

Module 2: Number Theory for Problem-Solving

Use 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.

Module 3: Optimizing Time & Space Using Sequential Storage

Coding 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.

Module 4: Non-Linear Data Structures

Applying 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.

Module 5: Problem Solving using Advanced Topics

CP 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.

List 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.

Targeted 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.

Text 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.

Project 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

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)

Course Code: CBD3400
Course Title: Fundamentals of Cloud Computing
L:T:P:C – 3:0:0:3
Prerequisite: Introduction to Big Data

Course Description

This course introduces the core concepts and architecture of cloud computing. It provides an overview of service models, deployment models, virtualization, resource provisioning, and security in cloud environments. Students will understand how cloud computing supports scalable, on-demand services for businesses and developers.

Course Objectives

— Understand the fundamentals and evolution of cloud computing
— Explore different cloud service and deployment models
— Learn about virtualization and resource management in the cloud
— Examine cloud security, billing models, and industry platforms

Course Outcomes

CO1 (Understand): Describe cloud computing architecture, characteristics, and models
CO2 (Analyze): Compare cloud service models and deployment strategies
CO3 (Apply): Identify appropriate virtualization and provisioning methods
CO4 (Apply): Analyze cloud security and pricing models for applications

Course Content (45 Hours Total)

Module 1: Introduction to Cloud Computing – 11 Sessions

History and evolution of cloud computing, Characteristics of cloud, Benefits and challenges, Cloud architecture and components, Use cases and industry adoption

Module 2: Cloud Service and Deployment Models – 11 Sessions

Service models: IaaS, PaaS, SaaS, Deployment models: Public, Private, Hybrid, Community, Case studies of service providers (AWS, Azure, GCP)

Module 3: Virtualization and Resource Management – 11 Sessions

Concept of virtualization, Types: CPU, Storage, Network virtualization, Hypervisors, Containerization (Docker), Resource provisioning and scheduling

Module 4: Security, Billing, and Industry Trends – 12 Sessions

Cloud security issues and solutions, Identity and access management (IAM), Data protection, Billing and pricing models, SLAs, Future of cloud: Edge, Serverless, Multi-cloud

Textbooks

T1: Rajkumar Buyya et al., *Mastering Cloud Computing*, McGraw Hill Education
T2: Thomas Erl et al., *Cloud Computing: Concepts, Technology & Architecture*, Prentice Hall

Reference Books

R1: Anthony T. Velte, Toby J. Velte, *Cloud Computing: A Practical Approach*, McGraw Hill
R2: Dan C. Marinescu, *Cloud Computing: Theory and Practice*, Morgan Kaufmann
R3: Michael Miller, *Cloud Computing: Web-Based Applications That Change the Way You Work*, Que Publishing
R4: Arshdeep Bahga, Vijay Madisetti, *Cloud Computing: A Hands-On Approach*, VPT

Web Resources

W1: https://aws.amazon.com/what-is-cloud-computing/
W2: https://azure.microsoft.com/en-in/resources/cloud-computing-dictionary/
W3: https://cloud.google.com/learn/what-is-cloud-computing
W4: https://www.ibm.com/cloud/learn/cloud-computing
W5: https://www.redhat.com/en/topics/cloud-computing

Course Code: CBD3401
Course Title: Distributed Computing with Hadoop
L:T:P:C – 3:0:0:3
Prerequisite: Introduction to Big Data

Course Description
This course provides an in-depth understanding of distributed computing concepts and practical implementation using the Hadoop ecosystem. It covers HDFS, MapReduce, YARN, and an introduction to tools such as Hive, Pig, and HBase. Emphasis is placed on scalable data processing and real-world big data applications.

Course Objectives
— Understand the principles of distributed computing and Hadoop architecture
— Learn how Hadoop handles large-scale data using HDFS and MapReduce
— Explore data processing tools in the Hadoop ecosystem
— Apply Hadoop tools to solve real-world big data problems

Course Outcomes
CO1 (Understand): Describe the architecture and components of Hadoop and distributed file systems
CO2 (Analyze): Compare Hadoop MapReduce with traditional data processing models
CO3 (Apply): Implement data processing tasks using MapReduce and HDFS
CO4 (Apply): Use Hadoop ecosystem tools for querying and analysis of big data

---

Course Content (45 Hours Total)

Module 1: Introduction to Distributed Computing and Hadoop – 11 Sessions
Basics of distributed systems, Challenges in distributed computing, Hadoop overview, HDFS architecture, Namenode, Datanode, HDFS operations, Fault tolerance and replication
Module 2: MapReduce Framework – 11 Sessions
MapReduce programming model, Job lifecycle, Input/output formats, Partitioning, Sorting, Combiner, Counters, Writing MapReduce jobs in Java / Python
Module 3: Hadoop Ecosystem and YARN – 11 Sessions
YARN architecture, Resource Manager, Node Manager, Job scheduling, Introduction to Hive (SQL on Hadoop), Pig (Scripting), HBase (NoSQL), Zookeeper, Sqoop and Flume basics
Module 4: Big Data Use Cases and Performance – 12 Sessions
Performance tuning in MapReduce, Hadoop configuration and administration basics, Real-time case studies (e.g., log analysis, recommendation engines), Introduction to Spark for Hadoop users

---

Textbooks
T1: Tom White, *Hadoop: The Definitive Guide*, O'Reilly Media, 4th Edition
T2: Chuck Lam, *Hadoop in Action*, Manning Publications

Reference Books
R1: Alex Holmes, *Hadoop in Practice*, Manning Publications
R2: Vignesh Prajapati, *Big Data Analytics with R and Hadoop*, Packt Publishing
R3: Jason Venner, *Pro Hadoop*, Apress
R4: Boris Lublinsky et al., *Professional Hadoop Solutions*, Wiley

---

Web Resources

W1: https://hadoop.apache.org
W2: https://hive.apache.org
W3: https://pig.apache.org
W4: https://hbase.apache.org
W5: https://spark.apache.org

Course Code: CBD3402
Course Title: Edge Computing & IoT Integration with Cloud
L:T:P:C – 3:0:0:3
Prerequisite: Introduction to Big Data

Course Description
This course explores the synergy between edge computing, the Internet of Things (IoT), and cloud technologies. It focuses on distributed data processing, IoT device integration, fog computing, edge analytics, and cloud-based orchestration. Students will understand architectures, platforms, and protocols used for scalable and real-time data management in smart environments.

Course Objectives
— Understand the architecture and fundamentals of IoT, edge, fog, and cloud computing
— Explore communication protocols, devices, and middleware for IoT-cloud integration
— Learn the role of edge and fog computing in reducing latency and enabling real-time analytics
— Apply cloud platforms for scalable IoT deployment and data orchestration

Course Outcomes
CO1 (Understand): Explain architectures and technologies involved in IoT-edge-cloud ecosystems
CO2 (Analyze): Compare edge, fog, and cloud computing paradigms for IoT integration
CO3 (Apply): Design real-time edge analytics frameworks using appropriate communication protocols
CO4 (Apply): Integrate IoT systems with cloud platforms for scalable data processing and control

Course Content (45 Hours Total)

Module 1: Introduction to IoT and Cloud Integration – 11 Sessions
Overview of IoT systems, IoT architecture layers, IoT sensors, actuators, and gateways, Cloud computing for IoT, Cloud platforms for IoT: AWS IoT, Azure IoT Hub, Google Cloud IoT

Module 2: Edge and Fog Computing – 11 Sessions
Definition and need for edge computing, Fog computing vs. cloud computing, Edge device architecture, Edge gateways, Edge-cloud orchestration, Use cases (smart cities, healthcare, manufacturing)

Module 3: Protocols and Middleware for IoT Integration – 11 Sessions
IoT protocols: MQTT, CoAP, HTTP, AMQP, LPWAN, Middleware and API management, Pub/Sub models, Real-time messaging, Data acquisition and preprocessing at edge nodes

Module 4: Edge Analytics and Deployment Frameworks – 12 Sessions
Edge AI and analytics using TinyML and TensorFlow Lite, Stream processing on edge devices, Docker and Kubernetes for IoT containerization, Security in IoT-cloud integration, Case studies and emerging trends

Textbooks
T1: Rajkumar Buyya & Satish Narayana Srirama, *Fog and Edge Computing: Principles and Paradigms*, Wiley
T2: Arshdeep Bahga, Vijay Madisetti, *Internet of Things: A Hands-on Approach*, VPT

Reference Books
R1: F. Wortmann & K. Flüchter, *Internet of Things: Technology and Value Creation*, Springer
R2: Pethuru Raj, Anupama C. Raman, *The Internet of Things: Enabling Technologies*, CRC Press
R3: Perry Lea, *Edge Computing: A Primer*, O'Reilly Media
R4: Andrew Minteer, *Analytics for the Internet of Things*, Wiley

Web Resources
W1: https://aws.amazon.com/iot/
W2: https://azure.microsoft.com/en-us/services/iot-hub/
W3: https://cloud.google.com/solutions/iot
W4: https://www.eclipse.org/mqtt/
W5: https://www.edge-computing.org

Course Code: CBD3403
Course Title: Cloud Storage & Data Management
L:T:P:C – 3:0:0:3
Prerequisite: Introduction to Big Data

Course Description
This course provides comprehensive knowledge on cloud-based data storage and management solutions. It covers storage architectures, distributed file systems, data lifecycle management, backup, archiving, and security. The course also focuses on cloud-native and hybrid storage models with applications in scalable and resilient enterprise systems.

Course Objectives
— Understand the principles and architecture of cloud storage systems
— Explore data management strategies including backup, recovery, and archiving
— Analyze performance, scalability, and consistency models in storage
— Implement cloud-native storage solutions for structured and unstructured data

Course Outcomes
CO1 (Understand): Explain cloud storage models, architectures, and technologies
CO2 (Analyze): Evaluate storage performance, redundancy, and availability strategies
CO3 (Apply): Use distributed file systems and object stores for cloud-based data management
CO4 (Apply): Design and manage data lifecycle and security in cloud environments

Course Content (45 Hours Total)

Module 1: Fundamentals of Cloud Storage – 11 Sessions
Introduction to cloud storage, Types: block, file, object storage, Cloud storage architectures (centralized vs distributed), Characteristics: durability, availability, scalability, Examples: AWS S3, Azure Blob, GCP Cloud Storage
Module 2: Distributed File Systems and Storage Services – 11 Sessions
HDFS architecture, Hadoop storage layers, Google File System (GFS), Amazon S3 internals, CephFS, GlusterFS, Data redundancy, replication strategies, Storage SLAs
Module 3: Data Management Techniques – 11 Sessions
Data lifecycle management, Tiered storage, Storage provisioning and pooling, Backup and disaster recovery, Snapshots, Versioning, Archival systems, Metadata management
Module 4: Storage Security and Monitoring – 12 Sessions
Data encryption (at rest and in transit), Identity and access control (IAM), Secure file sharing, Auditing, Logging, Storage cost management, Monitoring tools (CloudWatch, Azure Monitor), Case studies on hybrid storage management
Textbooks
T1: Rajkumar Buyya et al., *Mastering Cloud Computing*, McGraw Hill Education
T2: Greg Schulz, *Cloud and Virtual Data Storage Networking*, CRC Press

---

Reference Books
R1: Tom Clark, *Designing Storage Area Networks*, Pearson Education
R2: Robert Spalding, *Storage Networks: The Complete Reference*, McGraw Hill
R3: Larry Coyne et al., *Cloud Storage Forensics*, Syngress
R4: James E. Smith, Ravi Nair, *Virtual Machines: Versatile Platforms for Systems and Processes*, Morgan Kaufmann

---

Web Resources
W1: https://aws.amazon.com/s3/
W2: https://cloud.google.com/storage
W3: https://learn.microsoft.com/en-us/azure/storage/
W4: https://ceph.io
W5: https://docs.openstack.org/swift/

Course Code: CBD3404
Course Title: Cloud-Based Big Data Architecture & Optimization
L:T:P:C – 3:0:0:3
Prerequisite: Introduction to Big Data

Course Description
This course explores the design and optimization of scalable big data architectures in cloud environments. It covers architectural patterns, cloud-native services, storage and compute optimization, data processing frameworks, and monitoring techniques. Students will learn how to integrate and optimize big data pipelines across platforms like AWS, Azure, and Google Cloud.

Course Objectives
— Understand the components and patterns of cloud-based big data architectures
— Explore distributed processing and storage technologies in the cloud
— Learn techniques to optimize compute, storage, and network resources
— Apply monitoring, automation, and cost-optimization strategies for big data pipelines

---

Course Outcomes

CO1 (Understand): Explain the architecture and components of cloud-based big data systems

CO2 (Analyze): Evaluate cloud services and technologies for scalability, performance, and reliability

CO3 (Apply): Design optimized data pipelines using cloud-native services and frameworks

CO4 (Apply): Implement monitoring and cost-management strategies in big data architectures

---

Course Content (45 Hours Total)

Module 1: Cloud-Based Big Data Architecture – 11 Sessions

Cloud architecture for big data, Lambda and Kappa architectures, Storage layers (data lakes, warehouses), Processing layers (batch, stream), Messaging and ingestion (Kafka, Pub/Sub), Design patterns

---

Module 2: Distributed Processing Frameworks in Cloud – 11 Sessions

Apache Hadoop and Spark on cloud, AWS EMR, Azure HDInsight, GCP Dataproc, Serverless big data (AWS Lambda, Google Dataflow), Data integration and orchestration (Apache NiFi, Airflow)

Module 3: Optimization of Compute, Storage, and Network – 11 Sessions

Vertical vs. horizontal scaling, Auto-scaling, Spot instances, Storage formats (ORC, Parquet, Avro), Columnar vs row-based, Network I/O and caching, Data locality and partitioning

---

Module 4: Monitoring, Cost Optimization & Case Studies – 12 Sessions

Monitoring tools (CloudWatch, Azure Monitor, Stackdriver), Logging, Alerts, Cost estimation and reduction techniques, Billing models, Case studies on cloud-native big data solutions in finance, retail, and healthcare

---

Textbooks

T1: Mark Wilkins, *Architecting Cloud-Native Applications*, O'Reilly Media

T2: Pradeep Pasupuleti, *Big Data Analytics with Azure*, Apress

---

Reference Books

R1: Boris Lublinsky et al., *Professional Hadoop Solutions*, Wiley

R2: Janakiram MSV, *Architecting the Cloud*, Wiley

R3: Tom White, *Hadoop: The Definitive Guide*, O'Reilly Media

R4: Bill Wilder, *Cloud Architecture Patterns*, O'Reilly Media

---

Web Resources

W1: https://aws.amazon.com/big-data/

W2: https://cloud.google.com/solutions/big-data

W3: https://azure.microsoft.com/en-us/solutions/big-data/

W4: https://dataengineering.wiki

W5: https://towardsdatascience.com

Course Code: CBD3405

Course Title: Serverless Computing & Microservices in Cloud

L:T:P:C – 3:0:0:3

Course Description
This course introduces serverless architecture and microservices in cloud computing. It explores FaaS (Function-as-a-Service), container orchestration, service discovery, API gateways, and CI/CD pipelines. Students will learn to design and deploy lightweight, modular applications with improved scalability, maintainability, and cloud efficiency.

Course Objectives
— Understand the principles of serverless computing and microservices
— Learn function-based application development with cloud platforms
— Explore containerization, orchestration, and API management
— Design scalable and resilient microservice architectures in the cloud

Course Outcomes
CO1 (Understand): Describe the architecture and components of serverless computing and microservices
CO2 (Analyze): Evaluate platforms and design strategies for scalable service deployment
CO3 (Apply): Implement serverless functions and microservices using cloud-native tools
CO4 (Apply): Design CI/CD-enabled, containerized microservices with monitoring and scaling capabilities

Course Content (45 Hours Total)

Module 1: Introduction to Serverless Computing – 11 Sessions
Serverless paradigm overview, Benefits and challenges, FaaS overview, Serverless platforms (AWS Lambda, Azure Functions, GCP Cloud Functions), Event-driven execution, Stateless design principles

Module 2: Microservices Architecture – 11 Sessions
Introduction to monolithic vs. microservices architecture, Communication protocols (REST, gRPC), Service discovery, Load balancing, API Gateways, Design patterns for microservices

Module 3: Containers and Orchestration – 11 Sessions
Containerization using Docker, Docker Compose, Kubernetes architecture, Pods and Services, Scaling and replication, CI/CD pipelines with Jenkins/GitHub Actions for microservices deployment

Module 4: Monitoring, Security, and Case Studies – 12 Sessions
Service observability (logging, metrics, tracing), Tools (Prometheus, Grafana), Security best practices, Serverless security considerations, Cost optimization, Real-world use cases in retail, healthcare, and fintech

Textbooks
T1: Kasun Indrasiri, Danesh Kuruppu, *Serverless Integration Design Patterns with Azure*, Apress
T2: Sam Newman, *Building Microservices: Designing Fine-Grained Systems*, O'Reilly Media

Reference Books
R1: Mike Roberts & Martin Fowler, *Production-Ready Microservices*, O'Reilly Media
R2: Tim Wagner, *Programming AWS Lambda*, O'Reilly
R3: Brendan Burns, *Designing Distributed Systems: Patterns and Paradigms for Scalable, Reliable Services*, O'Reilly
R4: Gaurav Raje, *Serverless Architectures on AWS*, Packt Publishing

Web Resources
W1: https://docs.aws.amazon.com/lambda
W2: https://kubernetes.io/docs
W3: https://azure.microsoft.com/en-us/services/functions/
W4: https://microservices.io
W5: https://faas-and-furious.io

Course Code: CBD3406
Course Title: Introduction to Data Science & Big Data
L:T:P:C – 3:0:0:3
Prerequisite: Nil

Course Description
This course introduces the fundamental concepts of data science and big data analytics. It provides an overview of data collection, cleaning, exploratory analysis, visualization, and basic predictive modeling. It also introduces scalable processing using big data frameworks like Hadoop and Spark. The course emphasizes real-world applications of data science across domains.

Course Objectives
— Understand the data science process and big data ecosystem
— Explore data collection, preparation, and analysis techniques
— Learn basic machine learning methods and evaluation techniques
— Introduce big data technologies and their role in modern analytics

Course Outcomes
CO1 (Understand): Describe the data science life cycle and big data characteristics
CO2 (Analyze): Perform exploratory and statistical data analysis techniques
CO3 (Apply): Implement basic machine learning models for classification and prediction
CO4 (Apply): Demonstrate big data processing using Hadoop and Spark platforms

Course Content (45 Hours Total)

Module 1: Introduction to Data Science and Big Data – 11 Sessions
What is data science?, Data science workflow and roles, Characteristics of big data (5Vs), Types and sources of data, Introduction to tools: Python, R, Jupyter, Big data ecosystem overview

Module 2: Data Wrangling and Exploratory Analysis – 11 Sessions
Data cleaning, Handling missing data and outliers, Feature engineering, Descriptive statistics, Data visualization (histograms, boxplots, scatterplots), Correlation analysis

Module 3: Introduction to Machine Learning – 11 Sessions
Supervised vs. unsupervised learning, Linear regression, Logistic regression, k-Nearest Neighbors, Model evaluation: Accuracy, Precision, Recall, Confusion matrix

Module 4: Big Data Platforms and Applications – 12 Sessions
Hadoop ecosystem: HDFS, MapReduce, YARN, Introduction to Spark and PySpark, RDDs and DataFrames, Real-world applications of data science in business, healthcare, and IoT

Textbooks

T1: Joel Grus, *Data Science from Scratch: First Principles with Python*, O'Reilly Media
T2: Cathy O'Neil, Rachel Schutt, *Doing Data Science*, O'Reilly Media

Reference Books
R1: Vignesh Prajapati, *Big Data Analytics with R and Hadoop*, Packt Publishing
R2: Anil Maheshwari, *Data Analytics*, McGraw Hill
R3: Thomas W. Miller, *Modeling Techniques in Predictive Analytics*, Pearson
R4: Tom White, *Hadoop: The Definitive Guide*, O'Reilly

Web Resources
W1: https://www.kaggle.com/learn
W2: https://www.datasciencecentral.com
W3: https://spark.apache.org
W4: https://hadoop.apache.org
W5: https://www.datacamp.com

Course Code: CBD3407
Course Title: Feature Engineering & Model Selection
L:T:P:C – 3:0:0:3
Prerequisite: Nil

Course Description
This course covers the essential techniques for preparing data and selecting models in machine learning workflows. It introduces feature engineering methods such as encoding, transformation, dimensionality reduction, and feature selection. It also emphasizes model comparison, cross-validation, hyperparameter tuning, and performance evaluation strategies for supervised learning tasks.

Course Objectives
— Understand the role and importance of feature engineering in model development
— Apply preprocessing and transformation techniques to real-world data
— Explore model selection, tuning, and validation approaches
— Evaluate machine learning models using appropriate performance metrics

Course Outcomes
CO1 (Understand): Explain the concepts of feature engineering and its impact on model performance
CO2 (Analyze): Evaluate features using statistical and model-based selection techniques
CO3 (Apply): Perform dimensionality reduction and transformation on datasets
CO4 (Apply): Select, validate, and tune models for classification and regression problems

Course Content (45 Hours Total)

Module 1: Introduction to Feature Engineering – 11 Sessions
Types of features, Importance of features in machine learning, Handling missing values, Encoding categorical variables, Binning and bucketing, Variable transformation (log, Box-Cox), Feature scaling (normalization, standardization)

Module 2: Advanced Feature Construction and Selection – 11 Sessions
Polynomial and interaction features, Feature extraction (date/time, text), Feature selection methods: filter, wrapper, embedded, Recursive Feature Elimination (RFE), Feature importance using tree-based models

Module 3: Dimensionality Reduction Techniques – 11 Sessions
Curse of dimensionality, PCA (Principal Component Analysis), t-SNE, LDA, Autoencoders (overview), Choosing number of components, Visualizing high-dimensional data

Module 4: Model Selection and Validation – 12 Sessions
Train/test split, k-Fold Cross-validation, Leave-one-out CV, Hyperparameter tuning (Grid Search, Random Search), Bias-variance trade-off, Evaluation metrics: Accuracy, ROC-AUC, MAE, RMSE, Model comparison techniques

Textbooks
T1: Alice Zheng & Amanda Casari, *Feature Engineering for Machine Learning*, O'Reilly Media
T2: Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, O'Reilly Media

Reference Books
R1: Max Kuhn & Kjell Johnson, *Feature Engineering and Selection: A Practical Approach*, CRC Press
R2: Jason Brownlee, *Machine Learning Mastery with Python*, Machine Learning Mastery
R3: Gareth James et al., *An Introduction to Statistical Learning*, Springer
R4: Sebastian Raschka, *Python Machine Learning*, Packt Publishing

Web Resources
W1: https://scikit-learn.org
W2: https://www.kaggle.com/learn/feature-engineering
W3: https://towardsdatascience.com
W4: https://www.analyticsvidhya.com
W5: https://machinelearningmastery.com

Course Code: CBD3408
Course Title: Big Data-driven Business Intelligence
L:T:P:C – 3:0:0:3
Prerequisite: CBD1700 – Introduction to Big Data

Course Description
This course explores how big data technologies empower business intelligence (BI) systems for strategic decision-making. It covers data warehousing, OLAP, big data analytics, real-time dashboards, and visualization tools. Students will gain practical insights into the design, development, and application of BI solutions using big data platforms.

Course Objectives
— Understand the role of big data in modern business intelligence systems
— Learn data warehousing and OLAP techniques for analytical processing
— Explore big data analytics for descriptive, predictive, and prescriptive insights
— Apply visualization and reporting tools for real-time decision support

Course Outcomes
CO1 (Understand): Explain the principles of business intelligence and big data analytics
CO2 (Analyze): Evaluate data models, OLAP operations, and data integration in BI systems
CO3 (Apply): Design BI dashboards using big data visualization tools

CO4 (Apply): Develop insights from structured and unstructured data using BI pipelines

Course Content (45 Hours Total)

Module 1: Foundations of Business Intelligence – 11 Sessions
BI concepts and components, Role of data in decision-making, Data-driven organizations, BI architecture, Key performance indicators (KPIs), Data integration and ETL processes

Module 2: Data Warehousing and OLAP – 11 Sessions
Data warehouse architecture, Star and snowflake schemas, Fact and dimension tables, OLAP operations (slice, dice, drill-down), Data marts, ETL with big data sources

Module 3: Big Data Analytics in BI – 11 Sessions
Types of analytics: descriptive, predictive, prescriptive, Real-time analytics, Tools: Hadoop, Spark, Hive, Stream processing with Kafka/Spark Streaming, Use cases in retail, finance, healthcare

Module 4: BI Dashboards and Visualization – 12 Sessions
Data storytelling, BI tools (Power BI, Tableau, Google Data Studio), Real-time dashboards, Interactive reports, Custom KPIs, Case studies, Industry applications and trends

Textbooks
T1: David Loshin, *Business Intelligence: The Savvy Manager's Guide*, Morgan Kaufmann
T2: Ramesh Sharda, Dursun Delen, Efraim Turban, *Analytics, Data Science, & AI: Systems for Decision Support*, Pearson, 11th Edition

Reference Books
R1: Larissa T. Moss, S. Atre, *Business Intelligence Roadmap*, Addison-Wesley
R2: Michael Minelli et al., *Big Data, Big Analytics*, Wiley
R3: Galit Shmueli et al., *Data Mining for Business Analytics*, Wiley
R4: Cindi Howson, *Successful Business Intelligence*, McGraw Hill

Web Resources
W1: https://powerbi.microsoft.com
W2: https://www.tableau.com/learn
W3: https://datastudio.google.com
W4: https://www.kdnuggets.com
W5: https://towardsdatascience.com

Course Code: CBD3409
Course Title: Time Series Analysis & Forecasting
L:T:P:C – 3:0:0:3
Prerequisite: CBD1700 – Introduction to Big Data

Course Description
This course focuses on the analysis of time-dependent data for forecasting and decision-making. It covers fundamental and advanced techniques in time series analysis, including decomposition, smoothing, ARIMA models, machine learning approaches, and evaluation methods. Applications in business, finance, IoT, and climate analytics are explored.

Course Objectives

— Understand the nature and components of time series data
— Learn classical statistical methods and modern techniques for forecasting
— Develop models to predict future values based on historical patterns
— Apply time series methods in real-world domains using Python or R

Course Outcomes

CO1 (Understand): Explain key concepts and techniques used in time series modeling
CO2 (Analyze): Decompose time series into components and evaluate model assumptions
CO3 (Apply): Develop and validate forecasting models using ARIMA and machine learning techniques
CO4 (Apply): Interpret time series results and apply them to practical problems in various domains

---

Course Content (45 Hours Total)

Module 1: Introduction to Time Series Data – 11 Sessions (Understand)

Definition and types of time series, Components: trend, seasonality, cyclicity, randomness, Time series plots, Stationarity, Lag plots, Autocorrelation (ACF) and Partial ACF (PACF)

Module 2: Classical Forecasting Techniques – 11 Sessions (Analyze)

Moving averages, Exponential smoothing (SES, Holt's, Holt-Winters), Model selection with AIC/BIC, Forecast accuracy metrics (MAE, MSE, RMSE, MAPE)

Module 3: ARIMA and Seasonal Models – 11 Sessions (Apply)

AR, MA, ARMA, ARIMA models, Box-Jenkins methodology, Differencing, Seasonal ARIMA (SARIMA), Model diagnostics and residual analysis, Ljung-Box test

---

Module 4: Advanced and Machine Learning Approaches – 12 Sessions (Apply)

Vector Auto Regression (VAR), Facebook Prophet, LSTM for time series, Time series cross-validation, Feature engineering for time series, Use cases: sales, temperature, finance, energy

---

Textbooks

T1: Rob J. Hyndman, George Athanasopoulos, *Forecasting: Principles and Practice*, OTexts
T2: Chris Chatfield, *The Analysis of Time Series: An Introduction*, CRC Press

---

Reference Books

R1: Paul S.P. Cowpertwait, Andrew V. Metcalfe, *Introductory Time Series with R*, Springer
R2: Brockwell & Davis, *Time Series: Theory and Methods*, Springer
R3: Shumway & Stoffer, *Time Series Analysis and Its Applications*, Springer
R4: Jason Brownlee, *Deep Learning for Time Series Forecasting*, Machine Learning Mastery

---

Web Resources

W1: https://otexts.com/fpp3/
W2: https://www.statsmodels.org
W3: https://facebook.github.io/prophet/

W4: https://machinelearningmastery.com/time-series-forecasting/
W5: https://towardsdatascience.com

Course Code: CBD3410
Course Title: Natural Language Processing for Big Data
L:T:P:C – 3:0:0:3
Prerequisite: CBD1700 – Introduction to Big Data

Course Description
This course introduces core concepts and practical techniques in Natural Language Processing (NLP) applied to large-scale datasets. It covers text preprocessing, vector representations, syntactic and semantic analysis, sentiment classification, and neural models for text. Students will work with big data tools and NLP frameworks to extract, analyze, and visualize textual data from real-world sources.

Course Objectives
— Understand the fundamentals of NLP and linguistic data processing
— Explore algorithms for syntactic, semantic, and contextual analysis
— Apply NLP techniques on large-scale datasets using machine learning
— Use big data platforms to perform distributed text analytics and insights

Course Outcomes
CO1 (Understand): Explain foundational NLP techniques for text processing and representation
CO2 (Analyze): Evaluate NLP pipelines using linguistic and statistical models
CO3 (Apply): Implement sentiment analysis, topic modeling, and entity recognition on large datasets
CO4 (Apply): Use big data tools for scalable NLP workflows in real-world applications

Course Content (45 Hours Total)

Module 1: Fundamentals of NLP and Text Processing – 11 Sessions (Understand)
Text types and structures, NLP pipeline stages, Tokenization, Stop word removal, Lemmatization and stemming, POS tagging, Regular expressions, Bag-of-words and TF-IDF models

Module 2: Language Modeling and Syntax – 11 Sessions (Analyze)
N-gram models, Smoothing techniques, Syntax parsing (constituency and dependency), Named entity recognition (NER), Chunking, Word embeddings (Word2Vec, GloVe), Language resources (WordNet, corpora)

Module 3: NLP Applications on Big Data – 11 Sessions (Apply)
Text classification (Naïve Bayes, SVM), Sentiment analysis, Topic modeling with LDA, Text clustering, Text summarization, Document similarity and search, Case studies: product reviews, tweets, forums

Module 4: Scalable NLP with Big Data Tools – 12 Sessions (Apply)
Distributed text processing with Apache Spark (MLlib, Spark NLP), Text pipelines in spaCy, NLTK, and Hugging Face Transformers, Cloud-based NLP services (AWS Comprehend, Azure Text Analytics), Real-world project: NLP at scale

Textbooks

T1: Steven Bird, Ewan Klein, Edward Loper, *Natural Language Processing with Python*, O'Reilly
T2: Jacob Eisenstein, *Introduction to Natural Language Processing*, MIT Press

Reference Books
R1: Dan Jurafsky & James H. Martin, *Speech and Language Processing*, Pearson (3rd Ed. Draft)
R2: Yoav Goldberg, *Neural Network Methods for Natural Language Processing*, Morgan & Claypool
R3: Deepti Chopra, *Applied Natural Language Processing*, Packt Publishing
R4: Jason Brownlee, *Deep Learning for Natural Language Processing*, Machine Learning Mastery

Web Resources
W1: https://www.nltk.org
W2: https://spacy.io
W3: https://spark.apache.org/docs/latest/ml-guide.html
W4: https://huggingface.co/transformers/
W5: https://textblob.readthedocs.io

Course Code: CBD3411
Course Title: Data Governance, Ethics & Privacy
L:T:P:C – 3:0:0:3
Prerequisite: CBD1700 – Introduction to Big Data

Course Description
This course addresses the frameworks, principles, and practices for governing data responsibly. It focuses on data privacy, security, regulatory compliance, ethical AI practices, and responsible data usage in organizations. Students will explore legal aspects such as GDPR, data handling standards, and ethical dilemmas in data-driven decision-making.

Course Objectives
— Understand key concepts in data governance, ethics, and privacy
— Analyze regulatory frameworks, data handling policies, and global standards
— Explore ethical challenges in data science and AI applications
— Apply best practices for data protection and ethical compliance in projects

Course Outcomes
CO1 (Understand): Describe the principles of data governance, privacy, and ethical frameworks
CO2 (Analyze): Evaluate compliance requirements and legal aspects such as GDPR and data protection laws
CO3 (Apply): Develop policies and controls for ethical handling of sensitive and personal data
CO4 (Apply): Integrate data ethics and governance strategies into real-world data projects

Course Content (45 Hours Total)

Module 1: Introduction to Data Governance – 11 Sessions (Understand)
Principles of data governance, Data stewardship and ownership, Data policies, Metadata management, Data quality management, Governance frameworks (DAMA, COBIT), Data lifecycle

Module 2: Data Privacy and Protection Regulations – 11 Sessions (Analyze)
Global data protection laws: GDPR, CCPA, HIPAA, Consent management, Anonymization and pseudonymization, Privacy by design, Data breach reporting, Risk assessment and DPIA

Module 3: Ethics in Data Science and AI – 11 Sessions (Apply)
AI and algorithmic bias, Fairness, transparency, accountability in AI, Ethical data sourcing, Responsible AI principles, Case studies on ethical lapses and implications

Module 4: Implementing Governance and Ethics in Practice – 12 Sessions (Apply)
Data classification and access control, Policy enforcement and audits, Ethical codes for data professionals, Cloud data governance, Tools and platforms (Collibra, Apache Ranger), Real-world governance challenges

Textbooks
T1: Kord Davis, *Ethics of Big Data*, O'Reilly Media
T2: Michelle Dennedy, Jonathan Fox, Thomas Finneran, *The Privacy Engineer's Manifesto*, Apress

Reference Books
R1: Robert F. Smallwood, *Information Governance: Concepts, Strategies, and Best Practices*, Wiley
R2: David Martens, *Data Science Ethics*, Oxford University Press
R3: Joseph A. Schilling, *Data Governance: How to Design, Deploy and Sustain an Effective Data Governance Program*, TSO
R4: Viktor Mayer-Schönberger, Kenneth Cukier, *Big Data: A Revolution That Will Transform How We Live, Work, and Think*, Eamon Dolan/Houghton Mifflin Harcourt

Web Resources
W1: https://gdpr.eu
W2: https://www.privacy.org
W3: https://www.oecd.org/digital/privacy
W4: https://www.dama.org
W5: https://ethics.fast.ai

Course Code: CBD3412
Course Title: Computer Vision for AI Applications
L:T:P:C – 3:0:0:3
Prerequisite: Nil

Course Description
This course introduces the core concepts of computer vision and its applications in AI. It covers image formation, filtering, feature extraction, object detection, and deep learning models for visual recognition. Students will explore practical techniques to build intelligent vision systems using Python libraries and pre-trained models.

Course Objectives
— Understand the principles of image processing and computer vision
— Explore algorithms for feature extraction, object detection, and tracking
— Learn to apply deep learning models in visual recognition tasks
— Implement computer vision pipelines using real-world datasets and tools

Course Outcomes

CO1 (Understand): Explain basic image processing and vision principles
CO2 (Analyze): Evaluate computer vision algorithms for feature detection and representation
CO3 (Apply): Develop models for object detection, segmentation, and classification
CO4 (Apply): Use deep learning frameworks to build AI-enabled vision systems

Course Content (45 Hours Total)

Module 1: Introduction to Computer Vision – 11 Sessions (Understand)
Image formation and representation, Color models, Camera models, Sampling and quantization, Histogram equalization, Image filtering (smoothing, sharpening), Edge detection (Sobel, Canny)

Module 2: Feature Detection and Matching – 11 Sessions (Analyze)
Corner detection (Harris), Keypoint detectors (SIFT, SURF, ORB), Feature descriptors, Template matching, Homography and geometric transformations, Image stitching basics

Module 3: Object Detection and Tracking – 11 Sessions (Apply)
Object recognition pipelines, Face detection (Haar cascades), HOG+SVM, Object tracking (Mean-shift, KLT), Semantic segmentation overview, Contour detection, Background subtraction

Module 4: Deep Learning for Vision – 12 Sessions (Apply)
CNN architectures (LeNet, AlexNet, VGG, ResNet), Transfer learning, Image classification, Object detection using YOLO/SSD, Pre-trained models (OpenCV DNN, TensorFlow, PyTorch), Real-world applications (autonomous vehicles, surveillance, healthcare)

Textbooks
T1: Richard Szeliski, *Computer Vision: Algorithms and Applications*, Springer
T2: Adrian Rosebrock, *Deep Learning for Computer Vision with Python*, PyImageSearch

Reference Books
R1: Simon J. D. Prince, *Computer Vision: Models, Learning, and Inference*, Cambridge University Press
R2: Ian Goodfellow, Yoshua Bengio, Aaron Courville, *Deep Learning*, MIT Press
R3: Gary Bradski, Adrian Kaehler, *Learning OpenCV 4*, O'Reilly Media
R4: Mark Nixon, Alberto S. Aguado, *Feature Extraction and Image Processing for Computer Vision*, Academic Press

Web Resources
W1: https://opencv.org
W2: https://pyimagesearch.com
W3: https://pytorch.org/vision/stable/index.html
W4: https://tensorflow.org/tutorials
W5: https://paperswithcode.com

Course Code: CBD3413
Course Title: Reinforcement Learning for Big Data
L:T:P:C – 3:0:0:3
Prerequisite: Nil

Course Description

This course introduces the foundational concepts and algorithms in reinforcement learning (RL) with a focus on applications to big data environments. It explores value-based and policy-based learning, Markov decision processes, deep RL, and real-world applications in areas such as recommendation systems, robotics, and adaptive analytics.

Course Objectives

— Understand the principles and mathematical foundations of reinforcement learning
— Explore RL algorithms for decision-making under uncertainty
— Apply RL models in data-intensive environments using scalable tools
— Analyze real-world use cases of RL in business, healthcare, and AI systems

Course Outcomes

CO1 (Understand): Describe the components and goals of reinforcement learning systems
CO2 (Analyze): Evaluate different RL algorithms and their convergence behavior
CO3 (Apply): Implement RL algorithms for optimization and adaptive learning
CO4 (Apply): Use deep reinforcement learning techniques in big data applications

Course Content (45 Hours Total)

Module 1: Introduction to Reinforcement Learning – 11 Sessions (Understand)

Agent-environment interaction, Exploration vs exploitation, Markov Decision Processes (MDP), Bellman equations, Value function, Reward models, Policy evaluation

Module 2: Value-Based Learning – 11 Sessions (Analyze)

Dynamic programming, Monte Carlo methods, Temporal Difference (TD) learning, Q-learning, SARSA, Convergence and stability of value-based algorithms

Module 3: Policy-Based and Actor-Critic Methods – 11 Sessions (Apply)

Policy gradients, Stochastic policies, REINFORCE algorithm, Actor-critic models, Advantage functions, Trust region methods, Applications in sequential decision problems

Module 4: Deep Reinforcement Learning and Applications – 12 Sessions (Apply)

Deep Q Networks (DQN), Experience replay, Target networks, Proximal Policy Optimization (PPO), Scalable RL with TensorFlow and PyTorch, Use cases: personalized recommendations, financial trading, robotics, game AI

Textbooks

T1: Richard S. Sutton and Andrew G. Barto, *Reinforcement Learning: An Introduction*, MIT Press (2nd Edition)
T2: Maxim Lapan, *Deep Reinforcement Learning Hands-On*, Packt Publishing

Reference Books

R1: Csaba Szepesvári, *Algorithms for Reinforcement Learning*, Morgan & Claypool
R2: Yuxi Li, *Reinforcement Learning Explained*, Springer
R3: Ian Goodfellow, Yoshua Bengio, Aaron Courville, *Deep Learning*, MIT Press
R4: Alessandro Lazaric, *Reinforcement Learning and Dynamic Programming Using Function Approximators*, CRC Press

Web Resources

W1: https://www.davidsilver.uk/teaching/
W2: https://spinningup.openai.com
W3: https://github.com/dennybritz/reinforcement-learning

W4: https://pytorch.org/tutorials/intermediate/reinforcement_q_learning.html
W5: https://gym.openai.com

Course Code: CBD3414
Course Title: Generative AI and Large Language Models (LLMs)
L:T:P:C – 3:0:0:3
Prerequisite: Nil

Course Description
This course introduces the fundamental concepts and techniques of Generative AI with a focus on Large Language Models (LLMs). It explores language modeling, training methods, transformer architectures, ethical considerations, and practical applications in content generation, dialogue systems, and multimodal AI. Students will gain hands-on experience with open-source models and platforms.

Course Objectives
— Understand the architecture and working of large language models and generative AI
— Explore training methodologies, tokenization, and fine-tuning strategies
— Apply LLMs for various NLP tasks including generation, summarization, and translation
— Evaluate the ethical, social, and security concerns of using generative AI

Course Outcomes
CO1 (Understand): Explain the concepts, architectures, and workflows of generative AI and LLMs
CO2 (Analyze): Examine the performance and limitations of pre-trained LLMs across various NLP tasks
CO3 (Apply): Use open-source LLMs for real-world language generation and reasoning tasks
CO4 (Apply): Address ethical challenges and deploy LLMs responsibly in AI systems

Course Content (45 Hours Total)

Module 1: Introduction to Generative AI and LLMs – 11 Sessions (Understand)
Overview of generative models, Difference between discriminative and generative AI, Introduction to LLMs, Tokenization, Embeddings, Language modeling objectives, Applications of LLMs

Module 2: Transformer Architecture and Training – 11 Sessions (Analyze)
Self-attention and multi-head attention, Encoder-decoder and decoder-only models, Pretraining objectives (causal and masked language modeling), Fine-tuning strategies, OpenAI GPT, BERT, T5, LLaMA

Module 3: Prompt Engineering and Applications – 11 Sessions (Apply)
Few-shot, zero-shot, and chain-of-thought prompting, Prompt design strategies, Text generation, Summarization, Translation, Conversational AI, Code generation, Retrieval-augmented generation (RAG)

Module 4: Evaluation, Deployment, and Ethics – 12 Sessions (Apply)
Evaluation metrics (BLEU, ROUGE, perplexity), Hallucination and bias, Adversarial inputs, Security threats, Content moderation, Explainability in LLMs, Tools for deployment (LangChain, Hugging Face, OpenAI API), Responsible AI

Textbooks
T1: Sebastian Raschka et al., *Machine Learning with PyTorch and Scikit-Learn*, Packt (Ch. on Transformers)
T2: Eugene Charniak, *Introduction to Deep Learning and Language*, MIT Press

Reference Books
R1: Andrey Kurenkov, *The Illustrated Transformer* (online material)
R2: Tom B. Brown et al., *Language Models are Few-Shot Learners* (GPT-3 paper), arXiv
R3: Kevin Leyton-Brown, *CS324 – LLMs and Prompt Engineering* (Stanford Course Notes)
R4: Mikhail Pavlov et al., *Hugging Face Transformers*, O'Reilly Media

Web Resources
W1: https://huggingface.co
W2: https://platform.openai.com
W3: https://github.com/huggingface/transformers
W4: https://www.promptingguide.ai
W5: https://deepmind.com/research/publications

Course Code: CBD3415
Course Title: Explainable AI (XAI) and Ethical AI
L:T:P:C – 3:0:0:3
Prerequisite: Nil

Course Description
This course explores the emerging fields of Explainable Artificial Intelligence (XAI) and Ethical AI. It covers the need for transparency, interpretability, and trust in AI systems. Students will learn technical methods for explaining AI models, frameworks for ethical AI development, and guidelines for responsible deployment in sensitive domains such as finance, healthcare, and law.

Course Objectives
— Understand the importance of explainability and fairness in AI systems
— Learn technical methods for interpreting machine learning and deep learning models
— Analyze ethical challenges such as bias, accountability, and data privacy
— Apply frameworks and standards for building responsible and trustworthy AI solutions

Course Outcomes
CO1 (Understand): Explain the need for interpretability and fairness in AI systems
CO2 (Analyze): Evaluate different XAI techniques and their effectiveness across models
CO3 (Apply): Use XAI tools and libraries to explain AI model predictions
CO4 (Apply): Develop ethically aligned AI systems considering societal and legal norms

Course Content (45 Hours Total)

Module 1: Introduction to Explainable and Ethical AI – 11 Sessions (Understand)
Definition and need for XAI, Black-box vs. white-box models, Trade-offs in interpretability vs. performance, Overview of ethical AI principles: fairness, accountability, transparency

Module 2: Techniques for Explainable AI – 11 Sessions (Analyze)
Local and global interpretability, Feature importance, Surrogate models, LIME, SHAP, Partial Dependence Plots, Counterfactual explanations, Interpreting CNNs and attention mechanisms

Module 3: Tools and Platforms for XAI – 11 Sessions (Apply)
Open-source libraries: ELI5, SHAP, Alibi, InterpretML, Model cards and datasheets, Visualization dashboards, Case studies from healthcare, finance, and education

Module 4: Ethical and Responsible AI Practices – 12 Sessions (Apply)
Bias detection and mitigation, Privacy-preserving AI (differential privacy, federated learning), Legal frameworks (GDPR, AI Act), Guidelines from IEEE, OECD, UNESCO, Building trust and human-centric design

Textbooks
T1: Ankur Taly, Been Kim, *Practical Explainable AI using LIME and SHAP*, Packt Publishing, **2023**
T2: Virginia Dignum, *Responsible Artificial Intelligence: How to Develop and Use AI in a Responsible Way*, Springer, **2022**

Reference Books
R1: Christoph Molnar, *Interpretable Machine Learning*, Lulu Press (Online Free Edition), **2023**
R2: Mark Coeckelbergh, *AI Ethics*, MIT Press, **2022**
R3: Solon Barocas et al., *Fairness and Machine Learning: Limitations and Opportunities*, fairmlbook.org, **2023**
R4: Brent Mittelstadt & Luciano Floridi, *The Ethics of Artificial Intelligence*, Oxford Handbooks Online, **2022**

Web Resources
W1: https://www.shap.ai
W2: https://lime-ml.readthedocs.io
W3: https://www.fatml.org
W4: https://aix360.mybluemix.net – IBM AI Explainability Toolkit
W5: https://ethicsinaction.ieee.org

Course Code: CBD3416
Course Title: Advanced Deep Learning Architectures
L:T:P:C – 3:0:0:3
Prerequisite: Nil

Course Description
This course covers advanced neural network architectures and techniques used in solving complex AI tasks. Topics include CNNs, RNNs, GANs, Transformers, attention mechanisms, and hybrid models. Students will explore state-of-the-art deep learning frameworks and apply them to domains such as vision, NLP, and generative modeling.

Course Objectives
— Understand the design and training of advanced deep neural network architectures
— Explore attention-based models, recurrent and convolutional networks
— Apply deep generative and sequence models to real-world problems
— Evaluate architectural trade-offs and optimize performance across domains

Course Outcomes

CO1 (Understand): Describe advanced architectures used in deep learning systems
CO2 (Analyze): Compare the strengths and limitations of CNNs, RNNs, GANs, and Transformers
CO3 (Apply): Implement models using state-of-the-art libraries for vision and sequence tasks
CO4 (Apply): Fine-tune and optimize deep networks for performance and scalability

Course Content (45 Hours Total)

Module 1: Convolutional and Residual Networks – 11 Sessions (Understand)
Review of CNNs, Architectural innovations: AlexNet, VGG, ResNet, DenseNet, MobileNet, Transfer learning, Applications in computer vision

Module 2: Recurrent Networks and Sequence Modeling – 11 Sessions (Analyze)
RNNs, LSTMs, GRUs, Bidirectional RNNs, Sequence-to-sequence models, Attention in sequence modeling, Applications: speech, text, time series

Module 3: Generative Models – 11 Sessions (Apply)
Autoencoders (AEs, VAEs), Generative Adversarial Networks (GANs), DCGANs, Conditional GANs, Image generation, Anomaly detection, Text-to-image synthesis

Module 4: Transformers and Hybrid Architectures – 12 Sessions (Apply)
Transformer architecture and self-attention, BERT and GPT overview, Vision Transformers (ViT), Combining CNNs and RNNs, Training strategies and performance tuning, Application case studies

Textbooks
T1: Amandeep Singh, *Advanced Deep Learning with TensorFlow 2 and Keras*, Packt Publishing, **2023**
T2: Rowel Atienza, *Advanced Deep Learning with Python*, Packt Publishing, **2023**

Reference Books
R1: Ian Goodfellow, Yoshua Bengio, Aaron Courville, *Deep Learning*, MIT Press, **2022**
R2: Francois Chollet, *Deep Learning with Python*, Manning Publications, 2nd Ed., **2021**
R3: Ashish Vaswani et al., *Attention is All You Need*, NeurIPS, **2017**
R4: Jason Brownlee, *Generative Deep Learning Projects*, Machine Learning Mastery, **2023**

Web Resources
W1: https://keras.io
W2: https://pytorch.org
W3: https://huggingface.co/models
W4: https://paperswithcode.com
W5: https://www.tensorflow.org/tutorials

Course Code: CBD3417
Course Title: Real-Time Big Data Processing and AI Deployment
L:T:P:C – 3:0:0:3
Prerequisite: Nil

Course Description
This course introduces frameworks and tools for real-time big data stream processing and the deployment of AI models in production environments. It explores streaming

platforms like Apache Kafka and Spark Streaming, containerization, model serving, orchestration, and monitoring techniques to ensure scalable and reliable AI systems.

Course Objectives

— Understand real-time data processing architectures and tools
— Learn to build and deploy AI models in cloud and edge environments
— Explore CI/CD pipelines for AI workflows
— Apply monitoring and performance tuning techniques for production systems

Course Outcomes

CO1 (Understand): Describe real-time big data processing and AI deployment architectures
CO2 (Analyze): Compare stream processing frameworks and deployment strategies
CO3 (Apply): Build pipelines for real-time data analytics and AI model inference
CO4 (Apply): Deploy and monitor AI services using scalable production frameworks

Course Content (45 Hours Total)

Module 1: Introduction to Real-Time Data Processing – 11 Sessions (Understand)

Batch vs. stream processing, Real-time data characteristics, Apache Kafka, Apache Flink, Spark Streaming, Lambda vs. Kappa architecture, Event time processing, Use cases

Module 2: AI Model Deployment Fundamentals – 11 Sessions (Analyze)

Model serialization (Pickle, ONNX, SavedModel), Model serving frameworks (TensorFlow Serving, TorchServe), REST APIs with Flask/FastAPI, Deployment strategies: online, batch, shadow, blue-green

Module 3: Scalable Deployment with Containers and Orchestration – 11 Sessions (Apply)

Docker containers, Building and packaging ML apps, Kubernetes basics, Deploying AI services in Kubernetes, Helm charts, Scaling and resource allocation, Serverless inference

Module 4: CI/CD, Monitoring, and MLOps Tools – 12 Sessions (Apply)

CI/CD for AI (Jenkins, GitHub Actions), ML pipelines (MLflow, Kubeflow), Logging and monitoring (Prometheus, Grafana), Model drift detection, Alerting, Use cases in finance, IoT, healthcare

Textbooks

T1: Emmanuel Raj, *Machine Learning Engineering with Python*, Packt Publishing, **2023**
T2: Chi Wang & Donald Szeto, *Practical MLOps: Operationalizing Machine Learning Models*, O'Reilly Media, **2023**

Reference Books

R1: Jules S. Damji et al., *Streaming Systems with Apache Spark*, O'Reilly Media, **2022**
R2: Mark Treveil, *Kubeflow for Machine Learning*, O'Reilly Media, **2023**
R3: Noah Gift, *Practical MLOps*, Pearson, **2022**
R4: Holden Karau et al., *High Performance Spark*, O'Reilly, **2022**

Web Resources

W1: https://spark.apache.org/streaming/
W2: https://kafka.apache.org
W3: https://mlflow.org
W4: https://kubernetes.io
W5: https://cloud.google.com/vertex-ai

Course Code: **MAT2064**	**Course Title: Numerical Computing** **Type of Course:1] School Core**	**L-T- P-C**	**3**	**0**	**0**	**3**

Version No.		1.0
Course Pre-requisites		Calculus, Linear Algebra, Differential Equations
Anti-requisites		**NIL**
Course Description		The course explores mathematical techniques used to approximate solutions to complex problems that are difficult to solve analytically, often utilizing computers to perform calculations, including methods for root finding, interpolation, numerical differentiation and integration, solving systems of linear equations, and approximating solutions to differential equations, with applications across various scientific and engineering fields. It focuses on understanding the theoretical basis behind these methods, their implementation in programming languages, and analyzing their accuracy and stability.
Course Objective		The objective of the course is to equip students with understanding and ability to apply various numerical techniques to approximate solutions to complex mathematical problems that are difficult or impossible to solve analytically, particularly focusing on areas like solving systems of equations, finding roots of functions, interpolation, numerical differentiation, and integration, often utilizing computational tools to implement these methods.
Course Out Comes		On successful completion of the course the students shall be able to: CO1 - Calculate errors induced in the values by truncation of a series expansion. CO2 - Demonstrate the applications of numerical methods to find the roots of polynomial equations and eigen values of real symmetric matrices. CO3 - Apply the knowledge of numerical methods in modelling of various physical and engineering phenomena. CO4 - Apply various numerical methods for solving linear Ordinary & Partial differential equations arising in engineering field.
Course Content:		

Module 1	**Solution of Linear Systems of Equation**		**(12 Classes)**
Numerical Computation: Motivation and Objectives, Number Representation, Machine Precision, Round-of Error, Truncation Error, Random Number Generation.			

Solution of algebraic and transcendental equations: Various types of errors - Bisection method, Regula-Falsi method, Newton-Raphson method, Graffe's method - Bairstow's method - Newton's method for solving f(x,y) = 0 and g(x,y) = 0, secant method, Fixed point iteration method, Solution of linear system of equations, Gauss elimination method, Pivoting, Gauss Jordan method, Iterative methods of Gauss Jacobi and Gauss Seidel, Sufficient conditions for convergence - LU decomposition method, Eigenvalues of a matrix by Power method and Jacobi's method for symmetric matrices.

Module 2	**Interpolation and Approximation**	**Assignment**	**(8 Classes)**

Interpolation with equal intervals, Newton's forward and backward difference formulae, Interpolation with unequal intervals, Lagrange's interpolation, Newton's divided difference interpolation, Cubic Splines, Difference operators and relations.

Module 3	**Numerical Differentiation and Integration**		**(10 Classes)**

Numerical differentiation, Approximation of derivatives using interpolation polynomials, Numerical integration using Trapezoidal rule, Simpson's one-third rule, Simpson's three-eighth rule, Weddle's rule, Romberg's Method, Two point and three point Gaussian quadrature formulae, Evaluation of double integrals by Trapezoidal rule and Simpson's one-third rule

Module 4	**Initial & Boundary Value Problems for Ordinary & Partial Differential Equations**	**Assignment**	**(15 Classes)**

Single step methods — Taylor's series method, Modified Euler's method, Fourth order Runge-Kutta method for solving first order equations, Multi step methods, Milne's and Adams, Bash forth predictor corrector methods for solving first order equations.

Finite difference methods for solving second order, two-point linear boundary value problems, Finite difference techniques for the solution of two-dimensional Laplace's and Poisson's equations on rectangular domain, One-dimensional heat flow equation by explicit and implicit (Crank Nicholson) methods, One-dimensional wave equation by explicit method.

Targeted Application & Tools that can be used:
The contents of this course has direct applications in most of the core engineering courses for problem formulations, Problem Solution and system Design.
Tools Used: Python.

Assignment:

4. 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.

Text Book

1. C.F.Gerald and P.O.Wheatley", Applied Numerical Analysis", McGraw-Hill, 1981.
2. Cheneg and Kincaid, "Introduction to Numerical Computing", Tata McGraw-Hill, 1998.

References:

1. SRK Iyengar & RK Jain, Numerical Methods, New Age Internationals.
2. Erwin Kreyzig, Advanced Engineering Mathematics, John Wiley and sons, Inc.10th Edition
3. B. S. Grewal (2017), Higher Engineering Mathematics by, 44th Edition, Khanna Publishers.

E-resources/ Web links:

1. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_135224
2. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_141727
3. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_217628
4. http://.ac.in/courses.php?disciplineID=111
5. http://www.class-central.com/subject/math(MOOCs)
6. http://academicearth.org/
7. https://www.math.hkust.edu.hk/~maqian/ma006_0607F.html
8. https://www.scu.edu.au/study-at-scu/units/math1005/2022/

Topics relevant to SKILL DEVELOPMENT: The course focuses on the concepts of calculus and differential equation with reference to specific engineering problems. The course is of both conceptual and analytical type in nature through Problem solving**.** This is attained through the assessment component mentioned in course handout.

Course Code: CSE2514	**Course Title: Operating Systems** Lab **Type of Course: Lab Only**		**L-T- P- C**	0	0	2	1
Version No.	1.0					
Course Pre-requisites	CSE2009- Computer Organization 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 laboratory course provides hands-on experience with the core concepts of operating systems through practical assignments, simulations, and case studies. It covers foundational aspects such as system calls, process and thread management, inter-process communication, synchronization, deadlocks, memory management, and file systems. Students will implement and simulate real-time OS components and scheduling algorithms, fostering deeper understanding of OS architecture and design. The lab also introduces modern OS tools, programming interfaces, and the basics of open-source OS environments.					
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] Demonstrate system-level programming using system calls and OS structures. [**Apply**] 2] Simulate process scheduling and multithreading techniques. [**Apply**] 3] Apply various tools to handle synchronization problems using semaphores and shared memory. [**Apply**] 4] Demonstrate memory management and file system concepts using simulation or scripting. **[Apply]**					
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.

List of Laboratory Tasks:

Lab sheet -1

L1: Write a program to demonstrate the use of fork() and exec() system calls in process creation.

L2: A system has limited memory and high-priority real-time processes. Design a scheduling algorithm that ensures responsiveness while preventing starvation.

Lab sheet -2

L1: Implement First-Come-First-Serve (FCFS) process scheduling using C or Python.

L2: You are designing a server that handles thousands of client connections. Compare multithreading and multiprocessing for this task and implement a basic server model.

Lab sheet -3

L1: Implement Round Robin Scheduling with a fixed time quantum.

L2: In a banking system, concurrent access to accounts leads to data corruption. Design a synchronization solution to avoid race conditions.

Lab sheet -4

L1: Write a program to create threads using Pthreads or Python's threading module.

L2: You're tasked with building a file access tracker in an OS. Implement a system to log file access patterns and identify frequent accesses.

Lab sheet -5

L1: Demonstrate inter-process communication (IPC) using pipes.

L2: A simulation tool needs to emulate process suspension and resumption. Design and implement such a mechanism using signals or condition variables.

Lab sheet -6

L1: Simulate the Producer-Consumer problem using semaphores.

L2: You're developing a system where sensor devices (producers) generate temperature readings, and data processors (consumers) store and process

these readings. To prevent race conditions and ensure buffer safety, implement a synchronization mechanism using semaphores.

Lab sheet -7

L1: Implement Dining Philosophers Problem using threads and synchronization.

L2: In a multi-threaded cafeteria simulation, five philosophers sit around a circular table, each alternating between thinking and eating. To eat, a philosopher must hold two forks (represented by shared resources). Your task is to avoid deadlock and ensure no philosopher starves using thread synchronization techniques.

Lab sheet -8

L1: Write a program to simulate First Fit, Best Fit, and Worst Fit memory allocation strategies.

L2: A system with limited memory blocks needs to allocate memory to processes arriving with various size requests. Your task is to implement three classic memory allocation strategies—First Fit, Best Fit, and Worst Fit—to allocate memory to each process efficiently. Simulate and compare how memory gets allocated in each strateg

Lab sheet -9

L1: Demonstrate paging using a simple page table simulation.

L2: A program has a logical address space divided into pages. The system's memory is divided into equal-sized frames. When a program executes, its pages are loaded into available frames in main memory. Simulate the address translation process using a page table and demonstrate how a logical address is converted to a physical address.

Lab sheet -10

L1: Write a program to simulate page replacement algorithms like FIFO and LRU.

L2: In a virtual memory system, a process accesses pages in a specific order. The memory can only hold a limited number of pages (frames). When a page is needed and the memory is full, a page replacement algorithm is used to decide which page to evict. Simulate and compare FIFO and LRU algorithms for a given page reference string.

Lab sheet -11

L1: Simulate file directory structure (single level/two level).

L2: A university campus computer lab has limited memory space available for each student login session. When students open files or run programs, memory pages are loaded into available memory frames. Due to the limited number of frames, some pages must be replaced when new ones are needed. The lab system uses page replacement algorithms to decide which pages to evict when memory is full..

Lab sheet -12

L1: Write a shell script to demonstrate file handling commands in Linux.
L2: Design a command-line mini shell that can run background and foreground processes and handle basic built-in commands like cd, pwd, exit.

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

<table>
<tr><td>Course Code:

CSE1500</td><td colspan="3">Course Title: Computational Thinking Using Python

Type of Course: Integrated</td><td>L- T- P- C</td><td>2</td><td>0</td><td>2</td><td>3</td></tr>
<tr><td>Version No.</td><td colspan="8">1.0</td></tr>
<tr><td>Course Pre-requisites</td><td colspan="8">NIL</td></tr>
<tr><td>Anti-requisites</td><td colspan="8">NIL</td></tr>
<tr><td>Course Description</td><td colspan="8">This course introduces students to the essential skills of computational thinking and their practical application through the Python programming language. By combining problem-solving strategies with coding, students will learn to decompose complex challenges, identify patterns, abstract general principles, and design algorithms to build functional programs</td></tr>
<tr><td>Course Objective</td><td colspan="8">The objective of the course is to familiarize the learners with the concepts of Computational Thinking and use the Computational Thinking Principles to solve the computational Problems using Python Language</td></tr>
<tr><td>Course Outcomes</td><td colspan="8">Upon successful completion of this course, students will be able to:
• Explain and apply the core principles of computational thinking:
o Decomposition
o Pattern Recognition
o Abstraction
o Algorithm Design
• Use Python to implement solutions to real-world problems.
• Write and debug Python code using functions, loops and conditions
• Design simple programs and algorithms to automate repetitive or complex tasks.
• Collaborate effectively and communicate problem-solving approaches using pseudocode and Python.</td></tr>
<tr><td>Course Content:</td><td colspan="8"></td></tr>
<tr><td>Module 1</td><td colspan="3">Pillars of Computational Thinking</td><td colspan="2">Comprehension</td><td colspan="2"></td><td>9 Sessions</td></tr>
<tr><td colspan="9">What is computational thinking? Why is it important? Pillars of computational thinking: decomposition; pattern recognition; data representation and abstraction; algorithms

Applying computational thinking to case studies</td></tr>
</table>

Module 2	**Algorithm Design & Problem-Solving Strategies**	Application		**9 Sessions**
Introduction to Algorithms, Introduction to Problem Solving techniques: Brute Force, Divide and conquer, Common algorithms: find-max, linear search, binary search and other simple Algorithms				
Module 3	**Applied Computational Thinking using Python**	Application		**12 Sessions**
Introduction to Python, Data representation: variables, lists, Conditionals, Loops and Iteration Basic Example programs to illustrate the programming constructs				

Targeted Application & Tools that can be used:

Google Colab, Python

Text Book

1. "Computational Thinking for the Modern Problem Solver" – David D. Riley & Kenny A. Hunt
2. "Mastering Python 3 Programming: Ultimate Guide to Learn Python Coding Fundamentals and Real-World Applications" Subburaj Ramaswamy, BPB publications

References

1. • **Sweigart, Al.**
Automate the Boring Stuff with Python: Practical Programming for Total Beginners.
No Starch Press, 2015.
https://automatetheboringstuff.com

• **Severance, Charles.**
Python for Everybody: Exploring Data Using Python 3.
CreateSpace Independent Publishing, 2016.
https://www.py4e.com

• **Wing, Jeannette M.**
"Computational Thinking." *Communications of the ACM*, vol. 49, no. 3, 2006, pp. 33–35.
https://doi.org/10.1145/1118178.1118215

• **Downey, Allen B.**
Think Python: How to Think Like a Computer Scientist.

Green Tea Press, 2015.
http://greenteapress.com/wp/think-python-2e/

.

E-Resources

https://edu.google.com/resources/programs/exploring-computational-thinking

Topics relevant to "SKILL DEVELOPMENT": Decomposition, Abstraction, Pattern recognition, Data Representation ,Algorithms

Ittagalpura, Rajanukunte, Yelahanka, Bengaluru 560 119