
**Four-Year Undergraduate Programme**

**Bachelor of Technology**

**Computer Science and Engineering (CSE)**
**Faculty of Engineering & Technology**

**Parul University**

**Vadodara, Gujarat, India**

**Faculty of Engineering & Technology**
**Bachelor of Technology in Computer Science and Engineering**

## 1. Vision of the Department

To be recognized as a leading department for excellence in Computer Science & Engineering education and research, driving innovation and sustainable solutions that significantly contribute to societal advancement.

## 2. Mission of the Department

**M1** To deliver high-quality Computer Science education with strong theoretical, practical, and industry-relevant skills

**M2** To promote meaningful research through interdisciplinary collaboration and the effective use of emerging technologies.

**M3** To develop ethical, socially responsible professionals contributing to sustainable societal and technological progress.

## 3. Program Educational Objectives (PEOs)

The statements below indicate the career and professional achievements that the B.Tech. Computer Science and Engineering curriculum enables graduates to attain.

| | |
|---|---|
| **PEO 1** | To apply strong foundations in Computer Science to analyze, design, and develop effective solutions for real-world engineering problems. |
| **PEO 2** | To demonstrate professional ethics, effective communication, and teamwork skills to contribute responsibly in diverse professional environments. |
| **PEO 3** | To engage in lifelong learning, adapt to emerging technologies, and pursue innovation or research for sustainable professional and societal growth. |

## 4. Program Learning Outcomes

Program Learning outcomes are statements conveying the intent of a program of study.

| | | |
|---|---|---|
| **PLO 1** | **Engineering Knowledge:** | Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in **WK1 to WK4** respectively to develop to the solution of **complex engineering problems.** |

| | | |
|---|---|---|
| **PLO 2** | **Problem Analysis:** | Identify, formulate, review research literature and analyze **complex engineering problems** reaching substantiated conclusions with consideration for **sustainable development**. (**WK1 to WK4**) |

| | | |
|---|---|---|
| **PLO 3** | **Design/ Development of Solutions:** | Design creative solutions for ***complex engineering problems*** and design/develop systems/ components/ processes to meet identified needs with consideration for the **public health and safety, whole-life cost, net zero carbon, culture, society and environment** as required. **(WK5)** |
| **PLO 4** | **Conduct Investigations of Complex Problems:** | Conduct investigations of ***complex engineering problems*** using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. **(WK8).** |
| **PLO 5** | **Modern Tool Usage:** | Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve ***complex engineering problems***. **(WK2 and WK6)** |
| **PLO 6** | **The Engineer and Society:** | Analyze and evaluate societal and environmental aspects while solving ***complex engineering problems*** for its **impact on sustainability** with reference to economy, health, safety, legal framework, culture and environment. **(WK1, WK5, and WK7)**. |
| **PLO 7** | **Environment and Sustainability:** | Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. **(WK9)** |
| **PLO 8** | **Ethics:** | Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams |
| **PLO 9** | **Individual and Team work:** | Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences |

| | | |
|---|---|---|
| **PLO 10** | **Communication:** | Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments. |
| **PLO 11** | **Project Management and Finance:** | Recognize the need for, and have the preparation and ability for<br>i. independent and life-long learning<br>ii. adaptability to new and emerging technologies and<br>iii. critical thinking in the broadest context of technological change. **(WK8)** |

## 5. Program Specific Learning Outcomes (PSOs)

| | | |
|---|---|---|
| **PSO 1** | **Demand as per recent development** | Demonstrate the ability to analyze, design, verify, validate, code, and maintain efficient software solutions to address complex real-world problems using current trends and technological advancements in Computer Science and Engineering. |
| **PSO 2** | **Software skill** | Apply mathematical foundations, algorithmic principles, and core computing knowledge to analyze, design, and implement domain-specific solutions using modern tools and technologies across areas such as Intelligence, Data Science, Cloud Computing, and Cybersecurity. |

## 6. Credit Framework

| **Semester wise Credit Distribution of the Programme** |
|---|
| |

| **Category wise Credit distribution of the Programme** | |
|---|---|
| **Category** | **Credit** |

| | |
|---|---|
| Semester-1 | 16 |
| Semester-2 | 20 |
| Semester-3 | 22 |
| Semester-4 | 23 |
| Semester-5 | 22 |
| Semester-6 | 23 |
| Semester-7 | 27 |
| Semester-8 | 14 |
| **Total Credits:** | **167** |

| | |
|---|---|
| Major Core | 74 |
| Minor Stream | 0 |
| Multidisciplinary | 24 |
| Ability Enhancement Course | 9 |
| Skill Enhancement Courses | 9 |
| Value added Courses | 26 |
| Summer Internship | 16 |
| Research Project/Dissertation | 9 |
| **Total Credits:** | **167** |

## 7. Program Curriculum

| **Semester 1** | | | | | | |
|---|---|---|---|---|---|---|
| **Sr. No.** | **Subject Code** | **Subject Name** | **Credit** | **Lect** | **Lab** | **Tut** |
| 1 | 303192102 | Engineering Physics | 4 | 3 | 2 | 0 |
| 2 | 303191101 | Mathematics-I | 4 | 4 | 0 | 0 |
| 3 | 303105103 | Open-Source Software | 2 | 1 | 2 | 0 |
| 4 | 303193103 | Communication Skills | 2 | 0 | 0 | 2 |
| 5 | 303105104 | Computational Thinking for Structured Design 1 | 4 | 3 | 2 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 6 | 303104105 | Environmental Science | 0 | 1 | 0 | 0 |
| | | **Total** | **16** | **12** | **6** | **2** |

**Semester 2**

| Sr. No. | Subject Code | Subject Name | Credit | Lect | Lab | Tut |
|---|---|---|---|---|---|---|
| 7 | 303105151 | Computational Thinking for Structured Design 2 | 4 | 3 | 2 | 0 |
| 8 | 303107152 | ICT Workshop | 1 | 0 | 2 | 0 |
| 9 | 303105152 | Design Thinking | 3 | 2 | 2 | 0 |
| 10 | 303191151 | Mathematics-II | 4 | 4 | 0 | 0 |
| 11 | 303193152 | Advanced Communication & Technical Writing | 2 | 0 | 0 | 2 |
| 12 | 303105153 | Global Certifications- Fundamentals (AZ-900) | 2 | 2 | 0 | 0 |
| 13 | 303106103 | Electrical and Electronics Engineering | 4 | 3 | 2 | 0 |
| | | **Total** | **20** | **14** | **8** | **2** |

**Semester 3**

| Sr. No. | Subject Code | Subject Name | Credit | Lect | Lab | Tut |
|---|---|---|---|---|---|---|
| 14 | 303105201 | Design of Data Structures | 3 | 3 | 0 | 0 |
| 15 | 303105202 | Data Structure & Algorithms Laboratory | 2 | 0 | 4 | 0 |
| 16 | 303105203 | Database Management System | 3 | 3 | 0 | 0 |
| 17 | 303105204 | Database Management System Laboratory | 1 | 0 | 2 | 0 |
| 18 | 303105205 | Object Oriented Programming with JAVA | 2 | 2 | 0 | 0 |
| 19 | 303105206 | Object Oriented Programming with JAVA Laboratory | 1 | 0 | 2 | 0 |
| 20 | 303105220 | Digital Electronics | 3 | 3 | 0 | 0 |
| 21 | 303105221 | Digital Electronics Laboratory | 1 | 0 | 2 | 0 |
| 22 | 303191202 | Discrete Mathematics | 4 | 4 | 0 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 23 | 303193203 | Professional Communication Skills | 2 | 0 | 0 | 2 |
| | | **Total** | **22** | **15** | **10** | **2** |
| **Semester 4** | | | | | | |

| **Sr. No.** | **Subject Code** | **Subject Name** | **Credit** | **Lect** | **Lab** | **Tut** |
|---|---|---|---|---|---|---|
| 24 | 303105251 | Operating System | 3 | 3 | 0 | 0 |
| 25 | 303105252 | Operating System Laboratory | 1 | 0 | 2 | 0 |
| 26 | 303105210 | Computer Organization and Microprocessor | 3 | 3 | 0 | 0 |
| 27 | 303105211 | Computer Organization and Microprocessor Labs | 1 | 0 | 2 | 0 |
| 28 | 303105255 | Computer Network | 3 | 3 | 0 | 0 |
| 29 | 303105256 | Computer Network Laboratory | 1 | 0 | 2 | 0 |
| 30 | 303105257 | Programming in Python with Full Stack Development | 3 | 3 | 0 | 0 |
| 31 | 303105258 | Programming in Python with Full Stack Development Laboratory | 1 | 0 | 2 | 0 |
| 32 | 303191258 | Probability, Statistics and Numerical Methods | 4 | 4 | 0 | 0 |
| 33 | 303193252 | Professional Grooming and Personality Development | 1 | 0 | 0 | 1 |
| 34 | 303105259 | Competitive Coding | 2 | 0 | 4 | 0 |
| | | **Total** | **23** | **16** | **12** | **1** |
| **Semester 5** | | | | | | |
| **Sr. No.** | **Subject Code** | **Subject Name** | **Credit** | **Lect** | **Lab** | **Tut** |
| 35 | 303105218 | Design and Analysis of Algorithms | 3 | 3 | 0 | 0 |
| 36 | 303105219 | Design and Analysis of Algorithms Laboratory | 2 | 0 | 4 | 0 |
| 37 | 303105306 | Theory of Computation | 3 | 3 | 0 | 0 |
| 38 | 303105253 | Software Engineering | 3 | 3 | 0 | 0 |
| 39 | 303105254 | Software Engineering Laboratory | 1 | 0 | 2 | 0 |
| 40 | 303105309 | Enterprise Programming using Java | 2 | 2 | 0 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 41 | 303105310 | Enterprise Programming using Java Laboratory | 1 | 0 | 2 | 0 |
| 42 | 303105314 | Data Analytics and Data Visualization | 3 | 3 | 0 | 0 |
| 43 | 303105315 | Data Analytics and Data Visualization Laboratory | 1 | 0 | 2 | 0 |
| 42 | 303193304 | Professionalism & Corporate Ethics | 1 | 1 | 0 | 0 |
| 43 | | Open Elective 01 (Compulsory Subjects :1) | 2 | 2 | 0 | 0 |
| | | **Total** | **22** | **17** | **10** | **0** |

| **Open Elective 01** | | | | | | |
|---|---|---|---|---|---|---|
| **Sr. No.** | **Subject Code** | **Subject Name** | **Credit** | **Lect** | **Lab** | **Tut** |
| 1 | 303101331 | Basic Aircraft Science | 2 | 1 | 2 | 0 |
| 2 | 303105301 | AWS Fundamentals | 2 | 1 | 2 | 0 |
| 3 | 303105304 | Cyber Security | 2 | 1 | 2 | 0 |
| 4 | 303105305 | Internet of Things | 2 | 1 | 2 | 0 |
| 5 | 303107346 | Fundamentals of Communication Engineering | 2 | 1 | 2 | 0 |
| 6 | 303104305 | Disaster Preparedness and Planning | 2 | 2 | 0 | 0 |
| **Semester 6** | | | | | | |
| **Sr. No.** | **Subject Code** | **Subject Name** | **Credit** | **Lect** | **Lab** | **Tut** |
| 45 | 303105336 | Project – 1 | 3 | 0 | 6 | 0 |
| 47 | 303105311 | Quant, and Reasoning | 3 | 3 | 0 | 0 |
| 48 | 303105349 | Compiler Design | 3 | 3 | 0 | 0 |
| 49 | 303105350 | Compiler Design Laboratory | 1 | 0 | 2 | 0 |
| 50 | 303105341 | MEA(R)N Stack Web Development | 3 | 3 | 0 | 0 |
| 51 | 303105342 | MEA(R)N Stack Web Development Laboratory | 1 | 0 | 2 | 0 |
| 52 | | PEC 01(Compulsory Subjects: 1) | 3 | 3 | 0 | 0 |
| 53 | | PEC 01 – Labs-(Compulsory Subjects: 1) | 1 | 0 | 2 | 0 |
| 54 | | PEC 02 (Compulsory Subjects: 2) | 3 | 3 | 0 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 55 | | PEC 02 – Labs-(Compulsory Subjects: 2) | 1 | 0 | 2 | 0 |
| 56 | 303193353 | Employability Skills | 1 | 0 | 0 | 1 |
| | | **Total** | **23** | **15** | **14** | **1** |
| **PEC 01** | | | | | | |
| **Sr. No.** | **Subject Code** | **Subject Name** | **Credit** | **Lect** | **Lab** | **Tut** |
| 1 | 303105344 | Machine Learning | 3 | 3 | 0 | 0 |
| 2 | 303105307 | Artificial Intelligence | 3 | 3 | 0 | 0 |
| 3 | 303105363 | Cloud Computing | 3 | 3 | 0 | 0 |
| **PEC 01-LAB** | | | | | | |
| **Sr. No.** | **Subject Code** | **Subject Name** | **Credit** | **Lect** | **Lab** | **Tut** |
| 1 | 303105345 | Machine Learning Laboratory | 1 | 0 | 2 | 0 |
| 2 | 303105308 | Artificial Intelligence Laboratory | 1 | 0 | 2 | 0 |
| 3 | 303105364 | Cloud Computing Laboratory | 1 | 0 | 2 | 0 |
| **PEC 02** | | | | | | |
| **Sr. No.** | **Subject Code** | **Subject Name** | **Credit** | **Lect** | **Lab** | **Tut** |
| 1 | 303105379 | Mobile Application Development | 3 | 3 | 0 | 0 |
| 2 | 303105352 | .Net Programming | 3 | 3 | 0 | 0 |
| 3 | 303105354 | DevOps | 3 | 3 | 0 | 0 |
| **PEC 02-LAB** | | | | | | |
| 1 | 303105380 | Mobile Application Development Laboratory | 1 | 0 | 2 | 0 |
| 2 | 303105353 | .Net Programming Laboratory | 1 | 0 | 2 | 0 |
| 3 | 303105355 | DevOps Laboratory | 1 | 0 | 2 | 0 |
| **Semester 7** | | | | | | |
| **Sr. No.** | **Subject Code** | **Subject Name** | **Credit** | **Lect** | **Lab** | **Tut** |
| 57 | 303105431 | Summer Internship | 2 | 0 | 4 | 0 |
| 58 | 303105432 | Information and Network Security | 3 | 3 | 0 | 0 |
| 59 | 303105432 | Information and Network Security Laboratory | 1 | 0 | 2 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 60 | 303105433 | Project – II | 6 | 0 | 12 | 0 |
| 61 | 303105434 | Data Science | 3 | 3 | 0 | 0 |
| 62 | 303105435 | Data Science Laboratory | 1 | 0 | 2 | 0 |
| 63 | | PEC 03 (Compulsory Subjects: 1) | 3 | 3 | 0 | 0 |
| 64 | | PEC 03 – Labs-(Compulsory Subjects: 1) | 1 | 0 | 2 | 0 |
| 65 | | PEC 04 (Compulsory Subjects: 1) | 3 | 3 | 0 | 0 |
| 66 | | PEC 04 – Labs-(Compulsory Subjects: 1) | 1 | 0 | 2 | 0 |
| 67 | | Open Elective-2 | 3 | 3 | 0 | 0 |
| | | **Total** | **27** | **15** | **24** | **0** |

**Open Elective II**

| Sr. No. | Subject Code | Subject Name | Credit | Lect | Lab | Tut |
|---|---|---|---|---|---|---|
| 1 | 303105448 | Remote Sensing and Geo Informatics | 3 | 3 | 0 | 0 |
| 2 | 303105449 | Real Time Systems | 3 | 3 | 0 | 0 |
| 3 | 303105450 | Cyber Physical Systems | 3 | 3 | 0 | 0 |
| 4 | 303105451 | Computational Number Theory | 3 | 3 | 0 | 0 |
| 5 | 303105452 | VLSI System Design | 3 | 3 | 0 | 0 |

**PEC 03**

| Sr. No. | Subject Code | Subject Name | Credit | Lect | Lab | Tut |
|---|---|---|---|---|---|---|
| 1 | 303105485 | Augmented and Virtual Reality | 3 | 3 | 0 | 0 |
| 2 | 303105383 | Blockchain Technologies | 3 | 3 | 0 | 0 |
| 3 | 303105341 | Cyber Security | 3 | 3 | 0 | 0 |

**PEC 03-LAB**

| Sr. No. | Subject Code | Subject Name | Credit | Lect | Lab | Tut |
|---|---|---|---|---|---|---|
| 1 | 303105486 | Augmented and Virtual Reality Laboratory | 1 | 0 | 2 | 0 |
| 2 | 303105384 | Blockchain Technologies Laboratory | 1 | 0 | 2 | 0 |
| 3 | 303105342 | Cyber Security Laboratory | 1 | 0 | 2 | 0 |

**PEC 04**

| Sr. No. | Subject Code | Subject Name | Credit | Lect | Lab | Tut |
|---|---|---|---|---|---|---|
| 1 | 303105381 | Image Processing | 3 | 3 | 0 | 0 |
| 2 | 303105361 | Big Data Analytics | 3 | 3 | 0 | 0 |
| 3 | 303105355 | High performance computing | 3 | 3 | 0 | 0 |
| **PEC 04-LAB** | | | | | | |
| **Sr. No.** | **Subject Code** | **Subject Name** | **Credit** | **Lect** | **Lab** | **Tut** |
| 1 | 303105382 | Image Processing Laboratory | 1 | 0 | 2 | 0 |
| 2 | 303105362 | Big Data Analytics Laboratory | 1 | 0 | 2 | 0 |
| 3 | 303105356 | High performance computing Laboratory | 1 | 0 | 2 | 0 |
| **Semester 8** | | | | | | |
| **Sr. No.** | **Subject Code** | **Subject Name** | **Credit** | **Lect** | **Lab** | **Tut** |
| 68 | 303105499 | Internship | 14 | 0 | 28 | 0 |
| | | **Total** | **14** | **0** | **28** | **0** |
| | | **Total Credits** | **167** | | | |

**Semester 1 - 1**

a. **Course Name:** Engineering Physics

b. **Course Code:** 303192102

c. **Prerequisite:** Knowledge of Physics and some basic concepts in Mathematics like differentiation, integration, limit, differential equation, vector calculus up to 12th science level.

d. **Rationale:** Knowledge of physics is essential for all Engineering branches because physics is the foundation subject of all the branches of engineering and it develops the scientific temperament and analytical capability of engineering students.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Discuss the basics of quantum mechanics, including Schrodinger's equations and the physical significance of wave functions. |
| **CLOBJ 2** | Apply the Schrodinger equation to analyse particles in one-dimensional potential boxes, emphasizing practical implications and tunnelling effects. |
| **CLOBJ 3** | Master concepts of energy bands, semiconductor classification, E-k diagrams, and semiconductor device analysis including P-N junction diodes. |
| **CLOBJ 4** | Comprehensively understand material classification, focusing on magnetic materials, nanomaterials, and analysing physical, thermal, electrical, optical, and magnetic properties. |
| **CLOBJ 5** | Gain expertise in laser principles, types, and applications, as well as fiber optics principles and applications. Understand optoelectronic devices, their functionalities, and practical applications. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Explain the band structure and origin of band gap in semiconductors. |
| **CLO 2** | Formulate various theoretical aspects and the physical phenomena at atomic level. |
| **CLO 3** | Identify the materials useful for optoelectronic devices based on their optical transition processes. |
| **CLO 4** | Apply appropriate techniques to measure bandgap, resistivity, and other relevant parameters of semiconductors. |

| | |
|---|---|
| **CLO 5** | Explain the applications of low-dimensional semiconductor devices in modern electronics. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | - | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage** | **Teaching Hours** |
|---|---|---|---|
| 1 | **UNIT-I: Modern Physics**<br>Introduction about quantum Mechanics, Schrodinger's equations, Time dependent and Time Independent Wave Equation, Physical Significance of the wave Function, Application of Schrodinger equation in particles in One Dimensional Potential Box and Tunnelling effects. | 20% | 9 |
| 2 | **UNIT-II: Band Theory of Semiconductors**<br>Energy bands in solids, Classification of Materials into Semiconductors & Insulators, Density of state, E-k diagram, Kronig-Penny model (to introduce origin of band gap), and Effective mass. Direct and indirect band gap. Carrier Concentration in semiconductors, Fermi Level in Intrinsic and Extrinsic Semiconductors, P-N junction diode, Ohmic and Schottky Junction. | 20% | 9 |
| 3 | **UNIT-III: Materials**<br>Classification of materials: Magnetic materials, Nanomaterials based on semiconductors and metal oxides, Basic characteristic properties of nanomaterials, Novel Materi- also. Physical, Thermal, Electrical, Optical and Magnetic properties of materials. | 20% | 9 |

| | | | |
|---|---|---|---|
| 4 | **UNIT-IV: Laser and Fiber Optics**<br>Lasers: Interaction of radiation with Matter, Absorption, Spontaneous and Stimulated emission, Characteristics of Lasers, Types of Lasers: Ruby Laser, Helium-Neon Laser, Semiconductor Diode Laser, Applications of Lasers. Fiber Optics: Principle and Structure of Optical Fiber, Numerical Aperture of fibre, Types of Optical Fibers, Attenuation in Optical Fibers, Applications of Optical Fibers. | 20% | 9 |
| 5 | **UNIT-V: Devices**<br>Optoelectronic Devices: Photoconductive cell, photo-voltaic cell, Photodiode, Phototransistor, LED, IR emitter, Optocoupler, X-ray diffractometer, Quantum de- vices and their applications. | 20% | 9 |

**i. Text Books:**


1. J. Singh, Semiconductor Optoelectronics: Physics and Technology, McGraw- Hill Inc. (1995)
2. B. E. A. Saleh and M. C. Teich, Fundamentals of Photonics, John Wiley & Sons, Inc., (2007)
3. S. M. Sze, Semiconductor Devices: Physics and Technology, Wiley (2008)
4. Engineering Physics — HK Malek and A. K. Singh- McGraw Hill Publication
5. Semiconductor Optoelectronic Devices- P. Bhattacharya-Prentice Hall of India
6. Fundamentals of Physics- Halliday, Resnick and Walker


**j. List of Practicals:**

1. I-V characteristics of light emitting diode in forward bias.
2. I-V characteristics of Zener diode in reverse bias.
3. Determination of Velocity of ultrasonic waves in water.
4. Determination of Dielectric constants of Dielectric samples.
5. Measurement of Band gap of semiconductor material.
6. Measurement of Hall coefficient RH and carrier concentration in a semiconductor.
7. Measurement of Planck's constant using LED.
8. Measurement of wavelength of laser light using diffraction grating.
9. Measurement of Numerical aperture of an optical Fiber.
10. Moment of Inertia of a flywheel.
11. Measurement of power loss in an optical fibre.
12. B-H Curve tracing.

13. Determination of Young's modulus.
14. Determination of thermal conductivity. (Searle's method or Lee's method)

**Semester 1-2**

a. **Course Name:** Mathematics-I

b. **Course Code:** 303191101

c. **Prerequisite:** Knowledge of Mathematics up to 12th science level

d. **Rationale:** The Mathematics I syllabus integrates fundamental calculus concepts, advanced mathematical techniques, and matrix algebra, preparing students for engineering challenges with optimized problem-solving skills.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Develop a comprehensive understanding of definite and improper integrals, including the application of integration techniques to find areas and volumes in both Cartesian and Polar coordinates. |
| **CLOBJ 2** | Utilize differential equations to model and solve practical scenarios, demon-starting proficiency in various solution techniques. |
| **CLOBJ 3** | Analyze the convergence and divergence of sequences and series, employing tests such as the Alternating Series Test and Ratio Test. |
| **CLOBJ 4** | Analyze matrix operations and determinants, exploring their properties and applications in solving systems of linear equations. |
| **CLOBJ 5** | Apply Fourier series for representing periodic functions, verifying Dirichlet's conditions. |
| **CLOBJ 6** | Solve optimization problems using multivariable calculus concepts, such as Lagrange's multiplier. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Analyze various properties and forms of a matrix using its Eigen values and Eigen vectors. |
| **CLO 2** | Interpret the convergence of infinite sequence and series using various results and tests. |
| **CLO 3** | Discuss various properties such as limit, continuity, partial differentiability and applications of multivariate functions |
| **CLO 4** | Formulate mathematical model based on first order differential equations. |

| | |
|---|---|
| **CLO 5** | Evaluate area and volume using definite integrals and improper integrals |

**g. Teaching & Examination Scheme:**

| Teaching Scheme | | | | Evaluation Scheme | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 4 | - | - | 4 | 20 | 20 | - | 60 | - | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| Sr. No. | Topics | Weightage | Teaching Hours |
|---|---|---|---|
| 1 | **UNIT 1: Improper Integral & Application of Def- inite Integral**<br>Evaluation of definite and improper integrals, Beta and Gamma functions and their properties. Area bounded by curves in Cartesian and Polar form, Area of a region bounded by function, Area of a region bounded by curves in Parametric form, Volume by slicing, Volume of solid by revolution. | 8% | 5 |
| 2 | **UNIT 2: First Order Ordinary Differential Equation**<br>Exact, linear and Bernoulli's equations, Euler's equations, Equations not of first degree: equations solvable for p, equations solvable for y, equations solvable for x and Clairaut's type, Applications. | 15% | 9 |

| | | | |
|---|---|---|---|
| 3 | **UNIT 3: Matrices**<br>Matrices & Determinants with Properties, Linear Independence, Rank of Matrix, System of Linear Equations, Consistency of System, Solution of system of Linear Equations by Gauss Jordan and Gauss-Elimination Method, Eigenvalues, Eigenvectors, Symmetric, Skew-symmetric, and orthogonal Matrices, Eigenfaces, Diagonalization, Cayley Hamilton Theorem and its Applications, Diagonalization, Orthogonal Transformation, Quadratic form. | 25% | 15 |
| 4 | **UNIT 4: Sequences and Series**<br>Basics of Sequences, Bounded and Monotonic Sequences, Series, Convergence of sequence and series, Geometric series, P-series, Cauchy's Integral Test, Comparison Test, Alternating Series, Absolute and Conditional convergence, Ratio test, Cauchy's Root Test, Power series, Taylor's and Maclaurin's series. | 17% | 10 |
| 5 | **UNIT 5: Fourier Series**<br>Fourier Series of 2 periodic functions, Dirichlet's conditions for representation by a Fourier series, Fourier Series of a function of period 2, Fourier Series of even and odd functions, Half range series. | 10% | 6 |
| 6 | **UNIT 6: Multivariable Calculus (Differentiation)**<br>Functions of Several Variables, Limit, Continuity, Partial Derivatives, Homogeneous function, Euler's Theorem for homogeneous function, Modified Euler's Theorem, Chain Rule, Implicit function, Jacobian, Tangent plane and Normal line, Maximum and Minimum Values, Lagrange's Multiplier, Taylor's and Maclaurin's Series for functions of two variables. | 25% | 15 |

**i. Text Book and Reference Book:**


1. Calculus and Analytic Geometry (TextBook) By G.B. Thomas and R.L. Finney — Addison Wesley
2. Calculus with early transcendental functions By James Stewart — Cengage Learning
3. Higher Engineering Mathematics
   By B. S. Grewal — Khanna Publications
4. Elementary Linear Algebra (Text Book)
   By Howard Anton, Chris Rorres — Willy India Edition — 9th Edition
5. Advanced Engineering Mathematics (Text Book) By Erwin Kreyszig — Willey India Education
6. A textbook of Engineering Mathematics

By N.P. Bali and Manish Goyal — Laxmi Publications

**Semester 1-3**

a. **Course Name:** Open-Source Software

b. **Course Code:** 303105103

c. **Prerequisite:** Basic knowledge of software applications.

d. **Rationale:** Open Source has acquired a prominent place in software industry. Having knowledge of Open Source and its related technologies is an essential for Computer Science student. This course introduces Open-Source methodologies and ecosystem to students.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLO 1** | Gain familiarity with Principles of OSS, Open-Source Standards, requirements for Software, OSS success, Free Software, Examples, Licensing, Free Vs. Proprietary Software, Free Software Vs. Open-Source Software, Public Domain. |
| **CLO2** | Acquire Knowledge regarding Open-Source History, Open Source Initiatives, Open Standards Principles, Methodologies, Philosophy, Software freedom, Open-Source Software Development, Licenses, Copyright vs. Copy left, Patents, Zero marginal cost, Income-generation Opportunities, Internationalization |
| **CLO3** | Acquire knowledge of Community and Communication, Contributing to Open-Source Projects Introduction to GitHub, interacting with the community on GitHub, Communication and etiquette, testing open-source code, reporting issues, contributing code. Introduction to Wikipedia, contributing to Wikipedia or contributing to any prominent open-source project of student's choice. Open-Source Ethics and Social Impact: Open source vs. closed source, Open-source Government, Ethics of Open Source, |
| **CLO4** | Describe GNU/Linux, Android, Free BSD, Open Solaris. Open- Source Hardware, Virtualization Technologies, Containerization Technologies: Docker, Development tools, IDEs, Debuggers, Programming languages, LAMP, Open-Source Database technologies. |
| **CLO 5** | Demonstrate Apache Web server, BSD, GNU/Linux, Android, Mozilla (Firefox), Wikipedia, Drupal, WordPress, Git, GCC, GDB, GitHub, Open Office, Libre Office Study. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Differentiate between Open Source and Proprietary software and Licensing. |
| **CLO 2** | Recognize the applications, benefits, and features of Open-Source Technologies. |
| **CLO 3** | Demonstrate the ability to develop and manage open-source projects. |
| **CLO 4** | Evaluate the impact and importance of the open-source ecosystem in software development. |
| **CLO 5** | Apply open-source methodologies by studying relevant case studies and real-life examples. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 1 | 0 | 2 | 2 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage** | **Teaching Hours** |
|---|---|---|---|
| 1 | **UNIT-I: Introduction to Open-Source:**<br>Open Source, Need and Principles of OSS, Open-Source Standards, Requirements for Software, OSS success, Free Software, Examples, Licensing, Free Vs. Proprietary Software, Free Software Vs. Open-Source Software, Public Domain. History of free software, Proprietary Vs Open-Source Licensing Model, use of Open- Source Software, FOSS does not mean no cost. History: BSD, The Free Software Foundation and the GNU Project. | 15% | 1 |

| | | | |
|---|---|---|---|
| 2 | **UNIT-II: Open-Source Principles and Methodology**<br>Open-Source History, Open Source Initiatives, Open Standards Principles, Methodologies, Philosophy, Software freedom, Open-Source Software Development, Licenses, Copyright vs. Copy left, Patents, Zero marginal cost, Income- generation Opportunities, Internationalization.<br>Licensing: What Is A License, How to create your own Licenses, Important FOSS Licenses (Apache, BSD, PL, LGPL), copyrights and copy lefts, Patent. | 20% | 3 |
| 3 | **UNIT-III: Open-Source projects:**<br>Starting and maintaining own Open- Source Project, Open-Source Hardware, Open-Source Design, Open-source Teaching, and Open Source media. Collaboration: Community and Communication, Contributing to Open Source Projects Introduction to GitHub, interacting with the community on GitHub, Communication and etiquette, testing open-source code, reporting issues, con- tributing code. Introduction to Wikipedia, contributing to Wikipedia or contributing to any prominent open-source project of student's choice. Open-Source Ethics and Social Impact: Open source vs. closed source, Open-source Government, Ethics of Open source, Social and Financial impacts of open-source technology, Shared software, Shared source, Open Source as a Business Strategy. | 20% | 3 |
| 4 | **UNIT-IV: Understanding Open-Source Ecosystem:**<br>Open-Source Operating Systems: GNU/Linux, Android, Free BSD, Open Solaris. Open-Source Hardware, Virtualization Technologies, Containerization Technologies: Docker, Development tools, IDEs, Debuggers, Program- ming languages, LAMP, Open-Source Database technologies. | 20% | 4 |
| 5 | **UNIT-V: Case Studies**<br>Example Projects Apache Web server, BSD, GNU/Linux, Android, Mozilla (Firefox), Wikipedia, Drupal, Word-Press, Git, GCC, GDB, GitHub, Open Office, Libre Office Study: Understanding the developmental models, licensing, mode of funding, commercial/non- commercial use. | 25% | 4 |

**i. Text Books:**


1. Open-Source Technology", Kailash Vadera & Bhavyesh Gandhi, University Science Press, Laxmi Publications, 2009, Software Engineering, Sommerville (TextBook)

2. Open-Source Technology and Policy", Fadi P. Deek and James A. M. McHugh, Cambridge University Press, 2008 Software Engineering; Wiley India
3. Perspectives on Free and Open-Source Software", Clay Shirky and Michael Cusumano, MIT press.

**j. List of Practicals:**

1. Demonstration of Basic Linux commands.
2. Execute C Program using gcc compiler.
3. Demonstration of gprof command using Linux.
4. Create and Edit documents using Google Docs.
5. Create Presentation using Google Slides.
6. Demonstration of different Arithmetic and Logical Formulas using OpenOffice Calc.
7. Use of HTML to create simple web page.
8. Demonstration of MathML – a markup language for describing mathematical notation.
9. Demonstration of virtualization using Docker Container..
10. Demonstration GitHub Facility.

**Semester 1-4**

a. **Course Name:** Communication Skill

b. **Course Code:** 303193103

c. **Prerequisite:** Knowledge of English Language studied till 12th standard

d. **Rationale:** Basic Communication Skills are essential for all Engineers.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Gain familiarity with electrical current, potential difference, power and energy, sources of electrical energy and elements of electrical circuit. |
| **CLOBJ 2** | Solve problems related to Alternating current, alternating voltage, etc, Demonstrate a clear understanding of Pure R, L C circuit and combination of RLC, Series and Parallel combination of R, L and C, etc. |
| **CLOBJ 3** | Acquire knowledge of the resistor, capacitor, and inductor and their performance characteristics for series and parallel connections. |
| **CLOBJ 4** | Explain different single phase and three phase circuits. |
| **CLOBJ 5** | Demonstrate a clear understanding of the basic concepts, working principles and applications of transformer, DC machines and AC machines. |
| **CLOBJ 6** | Study the use of LT Switchgear, Fuse, MCB, ELCB etc. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Apply creative and critical thinking in academic and professional communication tasks. |
| **CLO 2** | Apply vocabulary with accurate pronunciation in academic and professional communication. |
| **CLO 3** | Use fundamental English grammar accurately in academic communication |
| **CLO 4** | Produce effective reading and writing responses for academic, professional, and social contexts. |
| **CLO 5** | Speak content clearly and appropriately in academic and professional settings, demonstrating growing confidence in their communication skills. |

**g. Teaching & Examination Scheme:**

| Teaching Scheme | | | | Evaluation Scheme | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| L | T | P | C | Internal Evaluation | | | ESE | | Total |
| | | | | MSE | CE | P | Theory | P | |
| 0 | 2 | 0 | 2 | 0 | 100 | 0 | 0 | 0 | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| Sr. No. | Topics | Weightage | Teaching Hours |
|---|---|---|---|
| 1 | **UNIT-I: Crazy Scientist:**<br>The students will be taught the importance of invention and innovation using some examples that changed the world the way it worked. | 5% | 2 |
| 2 | **UNIT-II: Phonetics**<br>IPA Introduction (listening racks), Phonic Sounds Pronunciation Practice including transcription. | 10% | 4 |
| 3 | **UNIT-III: Vocabulary Building & Word Formation Process**<br>Compounding, clipping, blending, derivation, creative re-spelling, coining and borrowing Prefixes & suffixes, synonyms & antonyms, standard abbreviations (related activities will be provided) . | 10% | 2 |
| 4 | **UNIT-IV: Speaking Activity: Role play on Critical Thinking (Life boat)**<br>This activity topic gears towards making students do role play based on various scenarios. It involves giving them a scenario and asking them to further develop the idea in a very interesting manner, then going on to enact it. It aims to improve students' convincing skills. | 10% | 4 |

| | | | |
|---|---|---|---|
| 5 | **UNIT-V: Picture Description & Picture Connector**<br>Enable students to use vocabulary and useful expression to describe the picture. In this class the students will be trained to form logical connections between a set of pictures which will be shared with them. This geared towards building creativity and presentation skills. | 15% | 2 |
| 6 | **UNIT-VI: Mine Activity: Usage of Preposition:**<br>Students will learn to use proper propositions by active participation in the activity. | 8% | 2 |
| 7 | **UNIT-VII: Worksheets on Identifying Common Errors in Writing:**<br>Sentence structure, Punctuations, Subject-Verb Agreement, Noun-Pronoun Agreement | 12% | 2 |
| 8 | **UNIT-V: Reading Skills**<br>The art of effective reading and its various strategies to be taught to the learners and practice exercises be given on reading comprehension. | 10% | 2 |
| 9 | **UNIT-IX: Speech and spoken Exchanges; Extempore:**<br>Students will learn the correct usage of spoken language as different from the written form. It will help the students in extempore speech. This will be done by making the students give variety of impromptu speeches in front of the class: 1 minute talk on simple topics. To change the average speakers in the class to some of the best Orator. | 10% | 4 |
| 10 | **UNIT-X: Book Review**<br>The learners will identify the central idea of the book, author's style and approach towards the book. This will enable the learners to express their point of view and hone their creativity and writing skills. | 10% | 4 |
| 11 | **UNIT-V: Activity Session**<br>This will enhance the creative thinking among students.<br>To develop their interpersonal communication skills. | 0% | 2 |

***Continuous Evaluation:** It consists of Assignments/Seminars/Presentations/Quizzes/ Surprise Tests (Summative/MCQ) etc.

**i. Text Books:**


1. Understanding and Using English Grammar Betty Azar & Stacy Hagen; Pearson Education.
2. Business Correspondence and Report Writing SHARMA, R. AND MOHAN, K.

3. Communication Skills Kumar S and Lata P; New Delhi Oxford University Press.
4. Technical Communication: Principles and Practice, Sangeetha Sharma, Meenakshi Raman; Oxford University Press.
5. Practical English Usage MICHAEL SWAN.
6. A Remedial English Grammar for Foreign Student F.T. WOOD.
7. On Writing Well, William Zinsser; Harper Paperbacks,2006; 30th anniversary edition.
8. Oxford Practice Grammar, John Eastwood; Oxford University Press.

**Semester 1-5**

a. **Course Name:** Computational Thinking for Structured Design-1

b. **Course Code:** 303105104

c. **Prerequisite:** Requires Basic Knowledge of Computer.

d. **Rationale:** This course is design to provide basic ideas of computer programming. This course also makes help to understand programming language. It will help to develop their logical abilities.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Explain programming basics and the fundamentals of C programming. |
| **CLOBJ 2** | Describe and use different data types in C. |
| **CLOBJ 3** | Apply mathematical and logical operations in C programs. |
| **CLOBJ 4** | Implement decision-making and looping constructs using if statements and loops. |
| **CLOBJ 5** | Organize and manipulate data using arrays in C. |
| **CLOBJ 6** | Demonstrate the use of pointers in C programming. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Describe the fundamental concepts of computers and their applications. |
| **CLO 2** | Explain the basic concepts of C programming language. |
| **CLO 3** | Design and develop various programming problems using C programming concepts. |

| | |
|---|---|
| **CLO 4** | Implement advance C programming concepts like function, pointer, structure and union etc. |
| **CLO 5** | Describe the file handling using C Programming language. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | - | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage** | **Teaching Hours** |
|---|---|---|---|
| 1 | **UNIT-I: Introduction to C language**<br>History of C language, Program Development Steps, Structure of C program | 10% | 3 |
| 2 | **UNIT-II: Data Types, User I/O and Operators**<br>**Data Types**<br>Extended and Derived Data types, Variables User I/O : Formatted, predefined Functions of stdio.h header file Operators: Types of operators, Precedence, Associativity. | 10% | 6 |
| 3 | **UNIT-III: Conditional Flow Statements:**<br>Iterative Statements, Jumping Statements and Pointers: Conditional Flow Statements: Simple if, if-else, else-if ladder, switch case Decision Making using conditional statements Iterative Statements: Control Entry and Control Exit Loops Jumping Statements: break, continue , forward and backward goto. Pointers: Typed: single double, triple. wild, NULL, Const, untyped, void. | 15% | 9 |

| | | | |
|---|---|---|---|
| 4 | **UNIT-IV: Functions:**<br>Functions :Call by value, call by references, Types of Functions. Pointer Functions: Calling A function through function pointer, Passing A function's address as an Argument to other function, Types of Pointer function Creation. Recursion : Types of Recursions : Direct Recursion, Indirect Recursion, Tail Recursion, No tail/Head Recursion, Tree Recursion, Nested Recursion. Storage classes: Auto, register, static and Extern. | 30% | 10 |
| 5 | **UNIT-V: Arrays:**<br>Arrays: Types of arrays, Declaration and Defining an array Pointer and Arrays: Types of Accessing Array elements Subscripting pointer variables Pointer to an ar- ray, Array of pointers, Pointers and two dimensional arrays Subscripting pointer To an array, Array of Functions : Strings: Strings v/s character arrays, Initializing strings, Reading and Displaying string Types of string for- mat Specifiers. puts() functions, Multi Line string Input String pointers, Two-dimensional character arrays or ar- ray of string Array of pointers to strings, String handling functions. | 35% | 14 |

**i. Text Books:**

1. C Programming by Bala Guru Swamy (TextBook)
2. C for all by S. Thammarai Selvi ,R Murugesan, Anuradha Publications.
3. programming in C Ajay Mittal, Pearson.

**j. List of Practicals:**

1. Installation C IDE, Basic Structure of C program. Format Specifiers, Escape Character. Run time input/Output Programs.
2. Write a c program to calculate Area of Rectangle, Perimeter of a Rectangle and Diagonal of a Rectangle.
3. The total distance travelled by vehicle in 't seconds is given by distance s = ut+1/2at? where 'u' and 'a' are the initial velocity (m/sec.) and acceleration(m/sec?). Write a C program to find the distance travelled at regular intervals of time given the values of 'u' and 'a'. The program should pro- vide the flexibility to the user to select his own time intervals and repeat the calculations for different values of 'u' and 'a'.
4. Write a C program to find the sum of individual digits of a positive integer.
5. A Fibonacci sequence is defined as follows: the first and second terms in the
6. Write a C program to find the roots of a quadratic equation.

7. Write C programs that use both recursive and non-recursive functions. 1. To find the factorial of a given integer.
8. To find the GCD (greatest common divisor) of two given integers.
9. Write a C program to find the largest integer in a list of integers,
10. Write a C program that displays the position or index in the string S where the string T begins, or -1 if S doesn't contain T
11. Write a C program to generate Pascal's triangle.
12. Write a C program to convert a Roman numeral to its decimal Equivalent.
13. Write a c program to take multiline string input and print individual string length .
14. Write a c program to reverse the individual word of a given string Explanation: input : Welcome To Bytexl output: emocleW oT lxetyB.

**Semester 1-6**

a. **Course Name:** Environmental Science

b. **Course Code:** 303104105

c. **Prerequisite:** Knowledge of Physics, Chemistry and Mathematics up to 12th science level and Biology up to 10th science level

d. **Rationale:** Basic knowledge of the environment is essential for all human beings for a good life and sustainable existence.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Apply systems thinking to analyze the city as a system, demonstrating application. |
| **CLOBJ 2** | Evaluate the role of smart citizens and approaches for citizen engagement. |
| **CLOBJ 3** | Identify sources and stressors of water resources, demonstrating under- standing. |
| **CLOBJ 4** | Analyze the causes, effects, and control measures of population explosion. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Explain the relationship between environmental health, ecology, and quality of life with reference to sustainable development. |

| | |
|---|---|
| **CLO 2** | Evaluate methods for pollution prevention and control considering various sources of pollution. |
| **CLO 3** | Assess the impacts of population growth on natural resources and environmental sustainability. |
| **CLO 4** | Examine recent technological and policy developments addressing various global environmental challenges. |
| **CLO 5** | Discuss the concept of smart cities and use of sustainable engineering solutions for urban development. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 1 | 0 | 0 | Audit | - | 50 | - | - | - | 50 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage** | **Teaching Hours** |
|---|---|---|---|
| 1 | **UNIT-I: ENVIRONMENTAL HEALTH, ECOL-OGY AND QUALITY OF LIFE**<br>Environmental education: Objective and scope, Impact of technology on the environment, Environmental disasters: Case studies, Global environmental awareness to mitigate stress on the environment, Structure and function of an ecosystem, Ecological pyramids, Pyramid of number, Pyramid of energy and pyramid of biomass. | 25% | 4 |

| | | | |
|---|---|---|---|
| 2 | **UNIT-II: POLLUTION PREVENTION**<br>Air & Noise pollution - Sources & their Effects, Case studies of Major Catastrophes, Structure and composition of the atmosphere, Water, Soil, Marine, Thermal & Marine Pollution: The story of fluoride contamination, Eutrophication of lakes, control measures, Measuring water quality: Water quality index, Waste water treatment (general) primary, secondary and tertiary stages, Municipal Solid waste management: Sources and effects of municipal waste, Biomedical waste, Hazardous waste. | 20% | 3 |
| 3 | **UNIT-III: POPULATION GROWTH, GLOBAL ENVIRONMENTAL CHALLENGES & LATEST DEVELOPMENTS**<br>Population Explosion - Causes, Effects and Control, an International initiative in population-related issues, Urbanization, Growth of the world's large cities, Water resources: Sources of water, Stress on water resources, Climate Change, Global Warming and Green House Effect, Acid Rain, Depletion of Ozone layer, Variation in concentrations of GHG gases in ambient air during last millennium, Role of Environmental Information System (EN- VIS) in India and similar programs run by EPA(USA), Role of soft tools like Quantum GIS, Autodesk Building Information Modelling (BIM) and City Finance Approach to Climate-Stabilizing Targets (C- FACT), Life Cycle Assessment, Bioinformatics and Optimization tools for sustainable development. | 25% | 4 |

| | | | |
|---|---|---|---|
| 4 | **UNIT-IV: SMART CITIES**<br>Introduction to smart cities - about smart cities, what is a smart city, world urbanization, case studies of Songdo, Rio De Janeiro, what makes cities smart.<br>City as a system of systems – Introduction, systems thinking, Milton Keynes Future Challenges, Rich picture as city challenges, Wicked problems, Development of smart city approach – core elements, open data, sustainability, privacy and ethics, development processes.<br>Smart Citizens – their role, engaging citizens, IES Cities, Energy systems, Approaches for Citizen Engagement, co-creating smart cities, cities unlocked, living labs, city problems, crowdsourcing ideas, redesigning cities for citizens, all age-friendly cities, mobility on demand, motion maps,<br>Infrastructure, Technology and Data – urban infrastructure and its technology, future of lighting, IoT, connected objects, sensing the city, NOx eating paints and air quality sensors, safest, smart citizen kit, sensing your city, Sen- sored City, Cyber security for data power, open, shared and closed data, satellite data, open data revolution, Smart City Project Data.<br>Innovation – smart innovations, smart city ecosystem, data-driven innovations for smart cities.<br>Standards and Capacity Building – the role of Standard, BSI smart city Standards, Hyper Cat, ITU Smart Sustain- able cities, Smart City Readiness, Lessons Learnt from Amsterdam.<br>Smart Measurements - metrics and indicators, city indicators, WCCD data portal, value proposition, integrated reporting, smart city learning and education, urban data school. | 30% | 6 |

**i. Text Books:**


1. Textbook of Environmental Studies For Undergraduate Courses (Text Book) By Dr Erach Bharucha — Orient BlackSwan — Second Edition, Pub. Year 2013.
2. Basics of Environmental Studies By U K Khare — Tata McGraw Hill.
3. Environmental Studies By Anindita Basak — Dorling Kindersley(India)Pvt. Ltd Pearson.
4. Environmental Sciences By Daniel B Botkin & Edward A Keller — John Wiley & Sons.
5. Air Pollution M N Rao , H .V N Rao — McGraw Hill Publishing Company Limited, New Delhi.

**Semester 2 - 1**

a. **Course Name:** Computational Thinking for Structured Design-2

b. **Course Code:** 303105151

c. **Prerequisite:** A foundational understanding of logic and problem-solving is a prerequisite for computational thinking in structured design.

d. **Rationale:** Computational thinking is integral for structured design as it fosters a systematic approach to problem-solving, breaking down complex issues into manageable components. By applying computational thinking principles, individuals can create well-organized and efficient structured designs, promoting clarity, maintainability, and scalability in software development. This methodology aligns with the logical and step by-step nature of structured design, enhancing the overall effectiveness of the development process.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Develop a deep understanding of foundational computational thinking concepts and their application in problem-solving. |
| **CLOBJ 2** | Demonstrate proficiency in creating structured designs using appropriate programming constructs and methodologies. |
| **CLOBJ 3** | Apply algorithmic thinking to decompose complex problems into manageable components, enhancing systematic problem-solving abilities. |
| **CLOBJ 4** | Evaluate and refine structured designs through critical analysis, promoting clarity, efficiency, and scalability in software solutions. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Develop proficiency in breaking down complex problems into manageable components, demonstrating a mastery of foundational computational thinking concepts. |
| **CLO 2** | Apply structured design principles to create efficient and well-organized algorithms, fostering a systematic approach to problem-solving in various domains. |
| **CLO 3** | Demonstrate the ability to design and implement structured programs using appropriate programming languages, showcasing practical skills in translating algorithms into executable code. |

| | |
|---|---|
| **CLO 4** | Cultivate a problem-solving mindset, emphasizing analytical thinking, algorithmic reasoning, and code optimization for developing scalable and maintainable software solutions. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | - | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage** | **Teaching Hours** |
|---|---|---|---|
| 1 | **UNIT-I Dynamic Memory Allocation:** malloc, calloc, realloc and free, Array of pointers, Programming Applications, Dangling Pointer | 10% | 6 |
| 2 | **UNIT-II Preprocessor Directives:** File Inclusion, Macros, Conditional Compilation and Pragmas. | 10% | 6 |
| 3 | **UNIT-III: Enumerators, Structures, Unions: Enumerators:** Enumerator Types **Structures:** Declaration Initialization Accessing Structures, Complex Structures, Structure and Functions Array of structures Arrays within structures Anonymous structures Nested structures pointers in structures Self-referential structures Structure Padding **Unions:** Bit fields **Typedef** | 15% | 15 |
| 4 | **UNIT-IV: Searching and Sorting:** Selection sort, Bubble Sort, Insertion sort, Quick sort and Merge Sort Linear and Binary Searching Techniques | 30% | 3 |
| 5 | **UNIT-V: Data Structures: List- Linear List: :** Singly Linked List - CRUD operations Double Linked List - CRUD operations Circular Linked List- CRUD operations | 35% | 15 |

**i. Text Books:**

1. Fundamentals of Data Structures in C, 2nd Edition, E. Horowitz, S, Sahni and Susan Anderson- Freed, Universities Press (Text Book)
2. Computer Programming & Data Structures - E. Balaguruswamy,4th Edition TMH
3. C & Data Structures - P . Padmanabham, Third Edition, B.S Publications
4. Classic Data Structures - D. Samanta

**j. List of Practicals:**

1. Write a c program to increase or decrease the existing size of an 1D array. 2. Write a c program on 2D array to Increase & Decrease i) No of subarrays ii) elements in the subarrays.
2. Write a to display present date and time using c language. 2. Write a c program to demonstrate pre-processor directives i) Macros ii) Conditional Compilation.
3. Write a C program that uses functions to perform the following Operations.
   i) Reading a complex number ii) Writing a complex number iii) Addition of two complex numbers iv) Multiplication of two complex numbers 2. Write a c program to store records of n students based on roll_no, name, gender and 5 subject marks i) Calculate percentage each student using 5 subjects. ii) Display the student list according to their percentages.
4. Write a C program to store n employee records based on EMP ID,EMP NAME,EMP DEPTID,EMP PHNO,EMP SALARY and display all the de- tails of employees using EMP NAME in sorted order.
5. Write a c program to implement selection Sort & Bubble sort 2. Write a C program to reverse the elements within a given range in a sorted list. Example : input : 10 9 1 2 4 3 4 6 7 8 10 3 8 output: 1 2 8 7 6 4 4 3 9 10 the sorted list of given array elements is 1 2 3 4 4 6 7 8 9 10 , after reversing the elements with in the range 3 and 8 is 1 2 8 7 6 4 4 3 9 10.
6. Write a c program to implement Insertion sort & Quick sort
7. Write a c program to sort the given n integers and perform following operations
   i) Find the products of every two odd position elements ii) Find the sum of every two even position elements Explanation: Input : 9 1 9 8 3 5 4 7 2 6 Output: 3 15 35 63 6 10 14 The sorted list of given input is 1 2 3 4 5 6 7 8 9, the product of alternative odd position elements is 1*3 = 3,3*5=15,5*7=35. . . and the sum of two even position elements 2+4 =6,4+6=10.
8. Write a C Program to implement Merge Sort.
9. Write a c program to sort in ascending order and reverse the individual row elements of an mxn matrix.

10. Write a c program to perform linear Search. 2. Write a c program to perform binary search.

11. Write a c program to Create a single Linked list and perform Following Operations A. Insertion At Beginning B. Insertion At End C. Insertion After a particular node D. Insertion Before a particular node E. Insertion at specific position F. Search a particular node G. Return a particular node H. Deletion at the beginning I. Deletion at the end J. Deletion after a particular node K. Deletion before a particular node L. Delete a particular node M. Deletion at a specific position.

12. Write a program to Reverse a singly Linked list. 2. Write a c program to check whether the created linked list is palindrome or not.

13. Write a c program to Create a Circular Linked list and perform Following Operations A. Insertion At Beginning B. Insertion At End C. Insertion After a particular node. D. Insertion Before a particular node E. Insertion at specific position F. Search a particular node G. Return a particular node H. Deletion at the beginning I. Deletion at the end J. Deletion after a particular node K. Deletion before a particular node L. Delete a particular node M. Deletion at a specific position

14. Write a c program to Create a Circular single Linked list and perform Following Operations A. Insertion After a particular node B. Insertion Before a particular node C. Search a particular node D. Return a particular node E. Deletion before a particular node F. Delete a particular node.

15. Write a c program to Create a Circular Double Linked list and perform Following Operations A. Insertion After a particular node B. Insertion Before a particular node C. Search a particular node D. Return a particular node E. Deletion before a particular node F. Delete a particular node.

**Semester 2 - 2**

a. **Course Name:** ICT workshop

b. **Course Code:** 303107152

c. **Prerequisite:** Basic Computer Knowledge and Physics

d. **Rationale:** This course is design to provide basic knowledge of Electronics components and computer components. This course helps in learning problem solving process of Electronics circuits and Computer.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Gain familiarity with identifying the Basic Electronic Components. |
| **CLOBJ 2** | Solve problems related to testing instruments such as Digital Multi meter, CRO , and function generator, etc. |
| **CLOBJ 3** | Acquire knowledge of Different sensors. |
| **CLOBJ 4** | Define and develop group projects using electronic components and sensors. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Explain the fundamentals of ICT, including internet, mobile computing, and problem-solving techniques using algorithms, flowcharts, and pseudocode. |
| **CLO 2** | Describe computer system components, software types, and the evolution of processors with their specifications. |
| **CLO 3** | Discuss computer architectures, CPU organization, and data representation using number systems and character encoding. |
| **CLO 4** | Differentiate volatile and non-volatile memories along with their characteristics and applications. |
| **CLO 5** | Identify components of data communication systems, transmission media, and types of signals with associated protocols. |
| **CLO 6** | Describe computer network types, topologies, models, and basic testing methods. |

g. **Teaching & Examination Scheme:**

| Teaching Scheme | | | | Evaluation Scheme | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| L | T | P | C | Internal Evaluation | | | ESE | | Total |
| | | | | MSE | CE | P | Theory | P | |
| 0 | - | 2 | 1 | - | - | 20 | - | 30 | 50 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. List of Practicals:**

1. Identification and symbolic representation of electronics basic components. (diode, zener diode, LED, transistor)
2. Verify the circuit analysis (voltage and current) using Digital Multimeter
3. Understanding of working and specifications of CRO and Function generator
4. Design 5V power supply using 7805.
5. Understanding soldering techniques and practicing proper soldering and de-soldering.
6. Demonstrate the working of Temperature Sensor
7. Verify the functionality of water flow sensor
8. Verify the functionality of distance measurement sensor
9. Demonstrate the working of Rain detector Sensor.
10. Group Project based on electronics components and sensors

**Semester 2 - 3**

a. **Course Name:** Design Thinking

b. **Course Code:** 303105152

c. **Prerequisite:** Open mindedness, curiosity, empathy, collaboration, iteration, creative thinking.

d. **Rationale:** Design thinking is a human cantered approach to problem-solving that emphasizes empathy, experimentation, and creativity. It is a framework for innovation and problem-solving that was originally developed in the context of product design but has since been applied to a wide range of fields and industries.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Develop a foundational understanding of design thinking, exploring its fundamentals and recognizing its significance in the context of product or service development. |
| **CLOBJ 2** | Master the skill of analyzing and interpreting the requirements of a given problem, gaining insights into the essential elements that drive effective problem-solving. |
| **CLOBJ 3** | Strategically plan and execute activities for problem resolution, employing techniques such as ideation and prototyping to foster innovative solutions within a structured framework. |
| **CLOBJ 4** | Hone the ability to evaluate proposed solutions, incorporating customer feedback into the iterative refinement process, ensuring that final out- comes align closely with user needs and preferences. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Discuss the basics of design thinking and its implications in product or service development. |
| **CLO 2** | Analyze the requirements of a typical problem. |
| **CLO 3** | Develop the necessary activities to solve a problem through ideation and prototyping. |
| **CLO 4** | Evaluate the solution and refine them based on the customer feedback. |

g. **Teaching & Examination Scheme:**

| **Teaching Scheme** | **Evaluation Scheme** |
|---|---|

| L | T | P | C | Internal Evaluation | | | ESE | | Total |
|---|---|---|---|---|---|---|---|---|---|
| | | | | MSE | CE | P | Theory | P | |
| 2 | - | 2 | 3 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| Sr. No. | Topics | Weightage | Teaching Hours |
|---|---|---|---|
| 1 | **UNIT-I Defining Needs, Ideation for solutions, Prototyping: Defining Needs:** Explain how PoV can be used in defining the design problem, Use a structured approach to arrive at a PoV. Ideation for Solutions: List the best practices for conducting a successful ideating session, Describe the techniques for evaluating and prioritizing ideas, Prototyping: Define prototyping, Explain how prototyping aids in communicating ideas effectively, List various tools for prototyping | 20% | 2 |
| 2 | **UNIT-II Testing the Solution, Problem Solving Mindset:** Testing the Solution: Define the steps of a successful testing approach, Demonstrate the process of gathering and responding to user feedback. Problem Solving Mind- set: Understanding Problem Statements, Recapping De- sign Principles, Design Thinking Toolsets, Formulating approaches to Solutions, Applications of Design Thinking: Case Study. | 20% | 2 |

| | | | |
|---|---|---|---|
| 3 | **UNIT-III: Human Centered Design, Design for the Environment:**<br>Human Centered Design: Services Development process and lifecycle, Product Vs Services, Innovation in Services, Service Experience Lifecycle, Human Computer Interaction, Usability Engineering - Heuristic Evaluation. De- sign for the Environment: Design Considerations, Environmental Issues, Sustainable Development, Green De- sign – Design for Process, Design for Product, Qualitative and Quantitative Methods for DFE, Design for Disassembly, Design for Recyclability, Design for Energy Efficiency. The relevance of 4Rs - reduction, reuse, recycling and recovery in Environmental friendly design. | 20% | 8 |
| 4 | **UNIT-IV: Design Thinking and Innovation Management Culture:**<br>Design Thinking and Innovation Management Culture: Project Management - Project Planning, Business Plan, Planning the resources, Effective Communication, Team Management, Benchmarking the Development, Cost Estimation, Interpreting the Feedback and Troubleshooting, Pitching the idea, Revenue Model. | 20% | 8 |
| 5 | **UNIT-V: Design Thinking and Innovation Management Culture:**<br>Design Thinking and Innovation Management Culture: Project Management - Project Planning, Business Plan, Planning the resources, Effective Communication, Team Management, Benchmarking the Development, Cost Estimation, Interpreting the Feedback and Troubleshooting, Pitching the idea, Revenue Model. | 20% | 8 |

**i. Text Books:**

1. The Design Thinking Playbook: Mindful Digital Transformation of Teams, Products, Services, Businesses and Ecosystems

**j. List of Practicals:**

1. **Introduction to design thinking:** Introduce the concept of design thinking, its benefits, and the overall process.
2. **Empathy mapping exercise:** Have participants conduct interviews with potential users and create empathy maps to gain a deeper understanding of their needs, wants, and pain points.
3. **Define the problem statement:** Based on the empathy mapping exercise, have participants synthesize their findings and define a problem statement.

4. **Ideation session:** Have participants generate as many ideas as possible to solve the problem statement. Encourage wild, unconventional, and innovative ideas.
5. **Prototyping session:** Have participants select one or more ideas and create a low-fidelity prototype to test their assumptions and validate their ideas.
6. **Testing and feedback session:** Have participants test their prototypes with potential users and gather feedback on what works, what doesn't, and what could be improved.
7. **Refine and iterate on prototype:** Based on the feedback, have participants refine and iterate on their prototype to improve its usability, functionality, and appeal.
8. **Presentation of final prototype:** Have participants present their final pro- to type to the rest of the group, explaining their design decisions, insights, and learnings.

**Semester 2 - 4**

a. **Course Name:** Mathematics-II

b. **Course Code:** 303191151

c. **Prerequisite:** Knowledge of Mathematics up to 12th science level

d. **Rationale:** The Mathematics I syllabus integrates fundamental calculus concepts, advanced mathematical techniques, and matrix algebra, preparing students for engineering challenges with optimized problem-solving skills.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Define and identify ordinary differential equations of higher order. Classify ODEs based on homogeneity and linearity. |
| **CLOBJ 2** | Solve homogeneous linear ODEs of higher order with constant coefficients, variable coefficients |
| **CLOBJ 3** | Apply the Method of Undetermined Coefficients to solve nonhomogeneous ODEs. |
| **CLOBJ 4** | Explain power series solutions for ordinary points and regular singular points. |
| **CLOBJ 5** | Define Laplace transform and its inverse. Explain the linearity property of Laplace transforms. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Solve higher-order ordinary differential equations with constant coefficients and Euler's ODE with variable coefficient |
| **CLO 2** | Use series solution methods and special functions like Bessel's' functions to solve the differential equations and analyze complex physical phenomena |
| **CLO 3** | Use the Laplace transform as tool to solve differential equations and Fourier integral representation |
| **CLO 4** | Apply Fourier integral to analyse the representation of functions in terms of frequency component. |
| **CLO 5** | Apply mathematical tools needed in evaluating vector calculus and their usage like Work, Circulation and Flux. |
| **CLO 6** | Use Multiple Integration technique to solve physical phenomena such as Area, Volume etc. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 4 | - | - | 4 | 20 | 20 | - | 60 | - | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage** | **Teaching Hours** |
|---|---|---|---|

| | | | |
|---|---|---|---|
| 1 | **UNIT 1: Higher order ordinary differential equations:** Ordinary differential equations of higher orders, Homogeneous Linear ODEs of Higher Order, Homogeneous Lin- ear ODEs with Constant Coefficients, Euler–Cauchy equations, Nonhomogeneous ODEs, Method of Undetermined Coefficients, Solution by Variation of Parameters, Applications | 8% | 5 |
| 2 | **UNIT 2 Power Series:** Power series solutions at ordinary point and regular singular point; Legendre polynomials, Bessel functions of the first kind and their property | 15% | 9 |
| 3 | **UNIT 3 Laplace Transform:** UNIT 3 Laplace Transform: Laplace Transform and inverse Laplace transform, Linearity, First Shifting Theorem (s-Shifting), Transforms of Derivatives and Integrals, ODEs, UNIT Step Function (Heaviside Function), Second Shifting Theorem (t-Shifting), Laplace transform of periodic functions, Short Impulses, Dirac's Delta Function, Convolution, Integral Equations, Differentiation and Integration of Transforms, Solution of ordinary differential equation by Laplace transform | 25% | 15 |
| 4 | **UNIT 4 Fourier Integral :** Fourier Integral, Fourier Cosine Integral and Fourier Sine Integral | 17% | 10 |
| 5 | **UNIT 5 Vector Calculus:** Gradient of scalar field, Directional Derivative, Divergence and curl of Vector field, Scalar line integrals, vector line integrals, scalar surface integrals, vector surface integrals, Theorems of Green, Gauss and Stokes. | 10% | 6 |
| 6 | **UNIT 6 Multivariable Calculus (Integration):** Multiple Integration: Double integrals (Cartesian), change of order of integration in double integrals, Change of variables (Cartesian to polar), Triple integrals (Cartesian) | 25% | 15 |

**i. Text Book and Reference Book:**


1. Advanced Engineering Mathematics (Textbook) By Erwin Kreyszig — Willey India Education
2. Calculus with early transcendental functions By James Stewart — Cengage Learning
3. Higher Engineering Mathematics By B. S. Grewal — Khanna Publications

**4.** Calculus and Analytic Geometry (Textbook) By G.B. Thomas and R.L. Finney — Addison Wesley A text book of Engineering Mathematics By N.P. Bali and Manish Goyal — Laxmi Publications

**Semester 2 - 5**

a. **Course Name:** Advanced Communication & Technical Writing

b. **Course Code:** 303193152

c. **Prerequisite:** Knowledge of English Language studied till 12th standard

d. **Rationale:** Communication confidence laced with knowledge of English grammar is essential for all engineers.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Gain familiarity with electrical current, potential difference, power and energy, sources of electrical energy and elements of electrical circuit. |
| **CLOBJ 2** | Solve problems related to Alternating current, alternating voltage, etc, Demonstrate a clear understanding of Pure R, L C circuit and combination of RLC, Series and Parallel combination of R, L and C, etc. |
| **CLOBJ 3** | Acquire knowledge of the resistor, capacitor, and inductor and their performance characteristics for series and parallel connections. |
| **CLOBJ 4** | Explain different single phase and three phase circuits. |
| **CLOBJ 5** | Demonstrate a clear understanding of the basic concepts, working principles and applications of transformer, DC machines and AC machines. |
| **CLOBJ 6** | Study the use of LT Switchgear, Fuse, MCB, ELCB etc. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Develop four basic skills |
| **CLO 2** | Construct grammatically correct sentences. |
| **CLO 3** | Develop and deliver professional presentation skills. |
| **CLO 4** | Develop the skills of critical thinking. |
| **CLO 5** | Compare different types of written communication. |

g. **Teaching & Examination Scheme:**

| Teaching Scheme | | | | Evaluation Scheme | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| L | T | P | C | Internal Evaluation | | | ESE | | Total |
| | | | | MSE | CE | P | Theory | P | |
| 0 | 2 | 0 | 2 | 0 | 100 | 0 | 0 | 0 | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| Sr. No. | Topics | Weightage | Teaching Hours |
|---|---|---|---|
| 1 | **UNIT-I: Developing Effective Listening Skills:** To help students understand the meaning and importance of good listening skills, learning the traits of being a good listener through activity and listening audio tracks.. | 10% | 2 |
| 2 | **UNIT-II: Error analysis:** To provide insights into the complicated processes of language development as well as a systematic way for identifying, describing and explaining errors. (Tenses, Voices, Reported speech) | 10% | 4 |
| 3 | **UNIT-III: Delivering different types of speeches:** Students will understand and use the different patterns for structuring speeches, Welcome / Introductory speech Vote of Thanks speeches, Farwell speeches . | 10% | 2 |
| 4 | **UNIT-IV: Professional Presentations** : Students will learn Combating stage fright, Preparing power point presentation Delivering PPT. | 10% | 5 |
| 5 | **UNIT-V: Essay writing** : Students will overcome the common pitfalls in the task of essay writing by understanding, Basics of Paragraph development and paragraph jumble, Types of essays, Characteristic features of essays, Guiding Principles. | 10% | 4 |
| 6 | **UNIT-VI: Reading Comprehension:** : Employing Different Reading Skills, Activity, Practice | 10% | 2 |
| 7 | **UNIT-VII: Project Proposal:** To equip students with the various elements required to prepare a winning proposal. | 5% | 2 |

| | | | |
|---|---|---|---|
| 8 | **UNIT-V: Misplaced Modifiers**<br>**S**tudents will understand how to place the improperly separated word, phrase or clause from the word it describes. | 5% | 1 |
| 9 | **UNIT-IX: Movie Review:**<br>A movie show followed by writing a review. To provide an exposure to students how to express their opinions about some film or documentary with unbiased and objective approach. | 10% | 2 |
| 10 | **UNIT-X: Narrative Writing:**<br>Narrative writing helps them explore different characters and settings. To help students clarify their thinking, and teach them to express that in writing in an organized way. | 5% | 2 |
| 11 | **UNIT-XI: Activity Session**<br>Process of writing, Order of writing, Final draft & checklist for reports, Sample reports, Memorandum, Letter report | 10% | 2 |
| 12 | **UNIT-XII: Critical Thinking**<br>Need, relevance and Significance of Critical Thinking, Logic in problem solving and decision making(activities),Moral Reasoning (Case Studies) | 5% | 1 |
| 13 | **UNIT-XIII: Activity Session (Presentation)**<br>An activity where the scene of a press conference is created in the class. Students are encouraged to ask sharp questions and in turn are invited to assume roles of famous personalities, thus answering the questions posed. | 0% | 1 |

***Continuous Evaluation:** It consists of Assignments/Seminars/Presentations/Quizzes/ Surprise Tests (Summative/MCQ) etc.

**i. Text Books:**


1. Business Correspondence and Report Writing SHARMA, R. AND MO- HAN, K.
2. Communication Skills Kumar S and Lata P; New Delhi Oxford University Press Practical English Usage MICHAEL SWAN
3. A Remedial English Grammar for Foreign Student F.T. WOOD\
4. On Writing Well William Zinsser; Harper Paperbacks,2006; 30th anniversary edition
5. Oxford Practice Grammar, John Eastwood; Oxford University Press Technical Communication : Principles And Practice Sangeetha Sharma, Meenakshi Raman; Oxford University Press


**Semester 2-6**

**1a . Course Name:** Global Certifications - Fundamentals (Azure)

b. **Course Code:** 303105153

c. **Prerequisite:** Possess a fundamental understanding of cloud computing concepts and services. Familiarity with basic networking principles and a working knowledge of operating systems is recommended.

d. **Rationale:** Azure provides a comprehensive cloud platform by Microsoft, offering scalable and flexible computing resources for businesses. With a vast array of services, Azure facilitates seamless deployment, management, and scaling of applications. Its global presence and integration with various tools make it a versatile and reliable choice for organizations seeking efficient cloud solutions.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Develop a foundational understanding of cloud computing principles, exploring key concepts such as virtualization, scalability, and resource provisioning. |
| **CLOBJ 2** | Gain familiarity with a diverse range of Azure services, enabling the ability to assess and leverage appropriate tools for different cloud-based scenarios. |
| **CLOBJ 3** | Acquire knowledge of Azure security features, including identity and access management, encryption, and compliance, to ensure the implementation of robust and secure cloud solutions. |
| **CLOBJ 4** | Define Azure Service Level Agreements (SLAs) and the lifecycle of Azure services, allowing for informed decision-making, efficient resource management, and adherence to service quality commitments. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Explain the fundamental principles of cloud computing, including its characteristics, deployment models, and service models. |
| **CLO 2** | Describe the various Microsoft Azure services and their roles in building and managing cloud-based solutions. |
| **CLO 3** | Define the key Azure security features, including identity management, data protection, and network security mechanisms. |

| | |
|---|---|
| **CLO 4** | Discuss Azure Service Level Agreements (SLAs) and explain the Azure service lifecycle, including deployment, monitoring, and maintenance of cloud services. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 2 | 0 | 0 | 2 | 20 | 20 | - | 60 | - | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage** | **Teaching Hours** |
|---|---|---|---|
| 1 | **Cloud Concepts:** Understanding cloud computing principles, such as the different types of cloud models (public, private, hybrid), infrastructure-as-a-service (IaaS), platform-as-a-service (PaaS), and software-as-a-service (SaaS). | 15% | 4 |
| 2 | **Azure Services:** Familiarity with the various Azure services and their common use cases. This includes services like Azure Virtual Machines, Azure App Services, Azure Storage, Azure Functions, Azure SQL Database, and more. | 20% | 6 |
| 3 | **Security, Privacy, Compliance, and Trust:** Knowledge of Azure security features, identity and access management, Azure Active Directory, data protection, compliance frameworks, and Azure governance methodologies. | 20% | 6 |
| 4 | **Azure Pricing and Support:** Understanding Azure subscription options, cost management, pricing models, and the different support options available to Azure customers. | 15% | 4 |

| | | | |
|---|---|---|---|
| 5 | **Azure SLA and Service Lifecycles:** Familiarity with Azure Service Level Agreements (SLAs) and the Azure service lifecycle, including planned maintenance, updates, and deprecation policies. | 30% | 10 |

**i. Reference Books:**


1. Microsoft Azure Fundamentals: Understanding Azure: by Michael Collier and Robin Shahan - 3rd Edition
2. Azure for Architects: Implementing cloud design, DevOps, containers, IoT, and serverless solutions on your public cloud: by Ritesh Modi - 2nd Edition
3. Exam Ref AZ-900 Microsoft Azure Fundamentals by Jim Cheshire - 2nd Edition


## Semester 2 - 7

**a. Course Name:** Electrical and Electronics Engineering

**b. Course Code:** 303106103

**c. Prerequisite:** Knowledge of Physics and Mathematics up to 12th science level.

**d. Rationale:** The course provides introductory treatment of the field of Electrical Engineering to the students of various branches of engineering.

**e. Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Master analysis techniques including Kirchhoff's laws, simplification methods, superposition, Thevenin's, and Norton's theorems for effective DC circuit analysis. |
| **CLOBJ 2** | Solve problems related to Alternating current, alternating voltage, etc, Demonstrate a clear understanding of Pure R, L C circuit and combi-nation of RLC, Series and Parallel combination of R, L and C, etc. |
| **CLOBJ 3** | Explain different single phase and three phase circuits. |
| **CLOBJ 4** | Learn diode behaviours, rectification techniques, and transistor functions as switches and amplifiers in electronic circuits. |
| **CLOBJ 5** | Define sensors and transducers, their applications, and differentiate between their types and functionalities in electronic systems. |

**f. Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Define basic concepts of various laws, principles and theorems associated with DC circuits for networks analysis. |
| **CLO 2** | Apply concepts of sinusoidal voltages, power relationships and show-casing knowledge of AC circuit theory using numerical and graphical representation. |
| **CLO 3** | Apply the fundamentals of diodes and transistors, including their characteristics, operation, and applications, to demonstrate understanding of electronics principles. |
| **CLO 4** | Design various types of voltage regulator circuits for power supply applications. |
| **CLO 5** | Classify various electronic sensors and transducers to understand their principles and real-world applications. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | - | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage** | **Teaching Hours** |
|---|---|---|---|
| 1 | **UNIT-I: DC Circuits**<br>Electrical circuit elements (R, L and C), voltage and current sources, Kirchhoff current and voltage laws, Mesh and Node analysis, Simplifications of networks using series and parallel combinations and star-delta conversions.<br>Superposition, Thevenin and Norton Theorems.. | 10% | 5 |

| | | | |
|---|---|---|---|
| 2 | **UNIT-II: AC Circuits**<br>AC Circuits Sinusoidal voltages and currents, their mathematical and graphical representation, Concept of instantaneous, peak (maximum), average and R.M.S. values, frequency, cycle, period, peak factor and form factor, phase difference, lagging, leading and in phase quantities and phasor representation. Rectangular and polar representation of phasors, pure inductance, pure capacitance and corresponding voltage- current phasor diagrams and wave- forms. Development of the concept of reactance, the study of series R-L, R-C, R-L-C circuit and resonance, study of parallel R-L, R-C and R-L-C circuit, concept of impedance, admittance, conductance and susceptance, the concept of active, reactive and apparent power and power factor,. Voltages, currents and power relations three-phase have balanced star-connected loads and delta-connected loads along with phasor diagrams. | 30% | 15 |
| 3 | **UNIT-III: Diode and Transistors**<br>Introduction to Ideal Diode, Effect of temperature Ideal diodes, unbiased diode and Forward and reverse bias of Diode. PIV, surge current, Diode as Uncontrolled switch. Rectifiers: Half wave, Full wave, and bridge wave. Ripple factor, PIV rating. Choke and Capacitor input filter rectifiers, Clipper and Clamper circuits, Voltage multiplier: Construction and working of BJT, Characteristics & specifications of BJT (PNP & NPN transistors), Biased and unbiased BJT, Configuration of the transistor, the concept of gain & BW, Operation of BJT in the cut-off, saturation & active regions (DC analysis), BJT as a switch, Transistor as an amplifier, Voltage divider bias and analysis, VDB load line and Q point. | 30% | 15 |
| 4 | **UNIT-IV: Voltage Regulator**<br>Lasers: Interaction of radiation with Matter, Absorption, Spontaneous and Stimulated emission, Characteristics of Lasers, Types of Lasers: Ruby Laser, Helium-Neon Laser, Semiconductor Diode Laser, Applications of Lasers. Fiber Optics: Principle and Structure of Optical Fiber, Numerical Aperture of fibre, Types of Optical Fibers, Attenuation in Optical Fibers, Applications of Optical Fibers. | 15% | 5 |
| 5 | **UNIT-V: Sensors and Transducers**<br>Introduction to sensors and Transducers, Comparison between sensors and Transducers, Applications of Sensors and Transducers, Types of Electronic sensors, Types of Transducers. | 15% | 5 |

**i. Text Books:**

1. A text book of Electrical technology Vol2, By B. L. Theraja — S. Chand Publication.
2. Electrical Engineering Fundamentals (Textbook), By V. D. Toro — Prentice Hall India — 2, Pub. Year 1989.
3. Electrical and Electronics Technology, By E. Hughes — Pearson — 10, Pub. Year 2010.
4. Basic Electrical Engineering , By D. P. Kothari and I. J. Nagrath, — Tata McGraw Hill — 3, Pub. Year 2010.
5. Basic Electrical Engineering, By D. C. Kulshreshtha — McGraw Hill — 1, Pub. Year 2009.
6. Fundamentals of Electrical Engineering, By Leonard S. Bobrow — Oxford University Press — 2, Pub. Year 1996.

**j. List of Practicals:**

1. To Study about Various Electrical and Electronics Symbols and demonstrate various measuring instruments used in Basic electrical Engineering laboratory.
2. To Perform and Solve Electrical Networks with Series and Parallel Combinations of Resistors Using Kirchhoff 's Laws.
3. To Obtain Inductance, Power and Power Factor of the Series RL Circuit With AC Supply Using Phasor Diagram.
4. To Obtain Capacitance, Power and Power Factor of the Series RC Circuit With AC Supply Using Phasor Diagram.
5. To Obtain Inductance, Capacitance, Power and Power Factor of the Series R-L-C Circuit With AC Supply Using Phasor Diagram.
6. Verification of superposition theorem with dc source.
7. Verification of Thevenin's theorem with dc source.
8. Verification of Norton's theorems in dc circuits.
9. Verification of Current and Voltage Relations in Three Phase Balanced Star and Delta Connected Loads.
10. To study the cut-section of a dc machine, single phase induction machine and three phase induction machine.
11. Find out the Efficiency and Voltage Regulation of Single Phase Transformer by Direct Load Test.
12. To Plot V-I characteristics Diodes. (a) PN junction diode Characteristics, (b) Zener Diode characteristics.
13. To Observe Rectifier Circuit (a) Half wave Rectifier without filter, (b) Full wave rectifier without filter, (c) Half wave

14. To Observe Response of Clipping and Clamping circuits using diodes (a) Diode Positive Clipper without and with Biased clipper, (b) Diode Negative Clipper without and with Biased clipper, (c) Biased Positive Negative Clipper (Combinational Clipper), and (d) Positive Clamper, and Negative Clamper.
15. Designing of power supply using IC regulator circuit. (a) Designing of +5 Volt DC Power Supply using 7805, (b) Designing of -5 Volt DC Power Supply using 7905, (c) Designing of +12 Volt DC Power Supply using 7812, and (d) Designing of -12 Volt DC Power Supply using 7912.
16. (a)To Plot and Study input-output characteristics of Common Base (B) configuration of the Transistor and (b) To Plot and Study input-output characteristics of common Emitter (CE) configuration of Transistor.
17. To study the Voltage divider bias circuit: (a) To observe the effect of change in base current on the Q-operating point, and (b) To set Q point for operation of a transistor amplifier in the linear region.
18. To plot characteristics of Schottky and Varactor diode.
19. Designing of Linear Adjustable Regulator using IC LM317.
20. Introduction to Sensors and Transducers.

SEMESTER -3

**Semester 3 - 1**

a. **Course Name:** Design of Data Structure

b. **Course Code:** 303105201

c. **Prerequisite:** Computer Programming and Basic Syntaxes

d. **Rationale:** Data structure is a subject of primary importance in Information and Communication Technology. Organizing or structuring data is important for implementation of efficient algorithms and program development. Efficient problem solving needs the application of appropriate data structure during program development.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Study the fundamental concepts of data structures**,** their classifications, operations, and algorithm complexity. |
| **CLOBJ 2** | Study the use of arrays, pointers, structures, and dynamic memory allocation for data representation. |

| | |
|---|---|
| **CLOBJ 3** | Apply the concepts of stacks, queues, and recursion to solve computational problems. |
| **CLOBJ 4** | Apply operations on linked lists including traversal, insertion, deletion, and searching. |
| **CLOBJ 5** | Apply various searching and sorting techniques for efficient data processing. |
| **CLOBJ 6** | Study and apply tree data structures including binary trees, binary search trees, AVL trees, and Red-Black trees. |
| **CLOBJ 7** | Study and apply hashing and graph traversal techniques such as Breadth First Search (BFS) and Depth First Search (DFS). |

**f. Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Define basic data structures, algorithm complexity, and searching techniques. |
| **CLO 2** | Apply stack and queue operations along with their algorithms to implement efficient applications. |
| **CLO 3** | Implement singly, doubly, and circular linked lists along with their applications in stacks and queues. |
| **CLO 4** | Apply tree data structures, including their algorithms and practical applications. |
| **CLO 5** | Implement sorting algorithms and hashing techniques in programs to organize and retrieve data. |
| **CLO 6** | Apply and analyze graph data structures, including their algorithms and practical applications. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | |
|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | **ESE** | |

| | | | | MSE | CE | P | Theory | P | Total |
|---|---|---|---|---|---|---|---|---|---|
| 3 | 0 | 4 | 5 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| Sr. No. | Topics | Weightage | Teaching Hours |
|---|---|---|---|
| 1 | **Introduction:** Data Structures, Classifications (Primitive & Non-Primitive), Data structure Operations, Re- view of Arrays, Structures, Self-Referential Structures, and Unions. Pointers and Dynamic Memory Allocation Functions. Representation of Linear Arrays in Memory, dynamically allocated arrays. Performance analysis of an algorithm and space and time complexities | 10% | 6 |
| 2 | **Stacks, Recursion and Queue:** Stacks: Definition, Stack Operations, Array Representation of Stacks, Stacks using Dynamic Arrays, Stack Applications: Polish notation, Infix to postfix conversion, evaluation of postfix expression. Recursion - Factorial, GCD, Fibonacci Sequence, Tower of Hanoi, Queues: Definition, Array Representation, Queue Operations, Circular Queues, Circular queues using Dynamic arrays, Deque, Priority Queues and its problems | 15% | 8 |
| 3 | **Linked Lists:** Definition, Representation of linked lists in Memory, Memory allocation; Garbage Collection. Linked list operations: Traversing, Searching, Insertion, and Deletion. Doubly Linked lists, Circular linked lists, and header linked lists. Linked Stacks and Queues. Applications of Linked lists | 10% | 5 |
| 4 | **Searching and Sorting:** Interpolation Search Sorts: Selection Sort, Insertion Sort, Bubble Sort, Quick Sort, Merge Sort, Radix Sort | 10% | 5 |
| 5 | **Trees:** Terminology, Binary Trees, Properties of Binary trees, Array and linked Representation of Binary Trees, Binary Tree Traversals - In Order, Post Order, Pre Order; Additional Binary tree operations. Threaded binary trees, Binary Search Trees – Definition, Insertion, Deletion, Traversal, Searching, Application of Trees-Evaluation of Expression. | 10% | 4 |

| | | | |
|---|---|---|---|
| 6 | **Red Black Trees and AVL Trees:** Introduction- Operations on Red Black Trees, AVL tree Construction, Operations on AVL Trees | 15% | 8 |
| 7 | **Hashing:** Hash Table organizations, Hashing Functions, Static and Dynamic Hashing | 15% | 3 |
| 8 | **Graphs:** Definitions, Terminologies, Matrix and Adjacency List Representation of Graphs, Elementary Graph operations, Traversal methods: Breadth First Search and Depth First Search. | 15% | 5 |

**i. Text Book and Reference Book:**

1. Fundamentals of Data Structures in C, 2ND EDITION, E.Horowitz, S,.Sahni and Susan Anderson- Freed, Universities Press (TextBook)
2. Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.

**j. List of Practical:**

1. Implement Stack and its operations like (creation push pop traverse peek search) using linear data structure
2. Implement Infix to Postfix Expression Conversion using Stack.
3. Implement Postfix evaluation using Stack.
4. Implement Towers of Hanoi using Stack.
5. Implement queue and its operations like enqueue, dequeue, traverse, search.
6. Implement Single Linked lists and its operations (creation insertion deletion traversal search reverse).
7. Implement Double Linked lists and its operations (creation insertion deletion traversal search reverse).
8. Implement binary search and interpolation search.
9. Implement Bubble sort, selection sort, Insertion sort, quick sort, merge sort.
10. Implement Binary Search Tree and its operations (creation, insertion, deletion).
11. Implement Traversals Preorder In-order Post-order on BST.
12. implement Graphs and represent using adjacency list and adjacency matrix and implement basic operations with traversals (BFS and DFS).

**Semester 3-2**

a. **Course Name:** Database Management System

b. **Course Code:** 303105203

c. **Prerequisite:** Basic Computer Knowledge

d. **Rationale:** The course will enable students to understand the different issues involved in the design and implementation of a database system as well execute various database queries using SQL.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Define DBMS and FPS |
| **CLOBJ 2** | Study the use of DBMS language, SQL |
| **CLOBJ 3** | Acquire knowledge of the different types of Model and E-R Diagram. |
| **CLOBJ 4** | Explain different Data Models, Constraints and keys, and relational algebra Model |
| **CLOBJ 5** | Study the use of transaction, database recovery, concurrency control and deadlock |
| **CLOBJ 6** | Study the use of Query Processing |
| **CLOBJ 7** | Define the different relational database designs |
| **CLOBJ 8** | Acquire knowledge of the security |
| **CLOBJ 9** | Explain the PL/SQL practical |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Explain database architecture, data models, and DDL/DML concepts. |
| **CLO 2** | Explain relational query languages, database design concepts, and query processing and optimization techniques. |
| **CLO 3** | Apply indexing, B-tree, and hashing techniques to optimize data storage and retrieval. |

| | |
|---|---|
| **CLO 4** | Analyze transaction processing and concurrency control techniques to ensure consistency, serializability, and reliable database recovery. |
| **CLO 5** | Evaluate database security mechanisms and access control models to select effective strategies against unauthorized access and attacks. |
| **CLO 6** | Assess advanced database systems, including object-oriented, distributed, web, and data warehousing solutions, to select optimal approaches for specific application requirements. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | 0 | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage** | **Teaching Hours** |
|---|---|---|---|
| 1 | **Introduction:** Introduction and applications of DBMS, File Processing System and its limitations, ANSI/SPARC Model, Data Independence, Client-Server Architecture, Users & DBA, Database Architecture. | 10% | 3 |
| 2 | **SQL:** Data Definition Language (DDL) commands, Data Manipulation Language (DML) commands, Data Control Language (DCL) commands, Transaction Control Language (TCL) commands. Predicates & Clauses: Logical Operators (AND / OR), Relational Operators, BE- TWEEN Predicate, IN & NOT IN Predicate, LIKE Predicate. Functions in SQL: Aggregate Functions, Character Functions, Arithmetic Functions, Date Functions, Conversion Functions. | 10% | 4 |

| | | | |
|---|---|---|---|
| 3 | **Data Models:** Hierarchical Model, Network Model, Relational Model, Object-Oriented Model. **E-R Diagram:** Introduction to E-R Diagram, Entities, Attributes & its types, Relationships, Mapping Cardinalities, Participation Constraints, Weak Entity Sets, Specialization, Generalization, Aggregation. | 10% | 5 |
| 4 | **Relational Data Model:** Introduction, Degree, Cardinality. **Constraints & Keys:** Primary Key, Foreign Key, Super Key, Candidate Key, Not Null Constraint, Check Constraint. **Relational Algebra Operations:** Selection, Projection, Cross-Product, Rename, Joins (Natural & Outer Join), Set Operators (Union, Intersection, Set Difference), Aggregate Functions. | 10% | 4 |
| 5 | **Relational Database Design:** Functional Dependency – definition, trivial and non-trivial FD, Armstrong's Axioms/Inference Rules, Closure of FD, Closure of Attributes, Candidate Key, Finding a Candidate Key, Decomposition (Lossy & Lossless), Database Anomalies, Normalization – 1NF, 2NF, 3NF, BCNF, 4NF, 5NF. | 20% | 6 |
| 6 | **Transaction:** Introduction, ACID Properties, Transaction Life Cycle, Scheduling, Serial Schedule, Interleaved Schedule, Transaction Operations, Serializability (View & Conflict), Two-Phase Commit Protocol. **Database Recovery:** Introduction, Log Based Recovery, Shadow Paging, Checkpoints. **Concurrency Control:** Introduction, Lock Based Protocol, Two Phase Lock Protocol, Intention Locking, Multiple Granularity, Time-based Protocol. **Deadlock:** Introduction, Deadlock Detection, Deadlock Recovery, Deadlock Prevention (Wait-Die, Wound-Wait & Timeout-Based Approach). | 20% | 12 |
| 7 | **Query Processing:** Introduction, Layers of Query Processing, Measures of Query Cost, File Scans (Linear & Binary Search), Materialized View, Pipelining. **Query Optimization:** Introduction, Equivalence Rules, Cost-Based Query Optimization. | 10% | 3 |
| 8 | **Security:** Data Security, Data Integrity, Authentication, Authorization, Encryption, Decryption, Access Control (DAC, RBAC, MAC), Intrusion Detection, SQL Injection. | 5% | 2 |
| 9 | **PL/SQL Concepts:** Views, PL/SQL Block, Cursors, Triggers, Stored Procedures, Stored Functions. | 5% | 3 |

**i. Text Books:**


1. Database System Concepts (TextBook) By Abraham Silberschatz, Henry Korth, S. Sudarshan — McGraw Hill International — 6th Edition

**2.** An Introduction to Database Systems By C. J. Date, A. Kannan, S. Swamynathan — Pearson Education

**3.** SQL, PL/SQL

List of Practical

1. Define Database Management System (DBMS) and explain the advantages of DBMS over the File Processing System
2. List 15 applications of database explain any two applications where databases are used.
3. Create a database for Student Details using MS Excel with appropriate fields.

4. Create a database for Employee Details using MS Access and insert suitable records.

5. Create a simple Facebook user database using MS Excel including fields like UserID, Name, Email, and Location.

6. Create tables in SQL using appropriate datatypes such as VARCHAR2, NUMBER, and DATE and insert sample data into the tables.

7. Write and execute simple SQL queries to display table structure and retrieve records using SELECT statement

8. SQL queries using conditions (WHERE clause) to retrieve records based on specific criteria.

9. Write SQL queries using LIKE operator and aggregate functions such as SUM, AVG, MAX, and COUNT.

10. Write and execute PL/SQL blocks to perform operations such as addition of two numbers, finding maximum numbers, calculating area, and generating Fibonacci series.

**Semester 3 - 3**

a. **Course Name:** Object Oriented Programming with JAVA

b. **Course Code:** 303105205

c. **Prerequisite:** Basic knowledge of software applications

d. **Rationale:** This course provides a broad introduction to software engineering. The various process models required to develop software are also described. Moreover, the functional and non-functional requirements are also described.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Gain the Knowledge of the concept with the Object-oriented programming, OOPs principles. |
| **CLOBJ 2** | Explain Data types, variables, operators. |
| **CLOBJ 3** | Define the concept of Control statements. |
| **CLOBJ 4** | Demonstrate the use of Arrays, Array values, and memory storage Structure. |
| **CLOBJ 5** | Demonstrate the use of various OOPs concepts with the help of programs. |
| **CLOBJ 6** | Study the use of Inheritance with Examples. |
| **CLOBJ 7** | Explain the concept of Strings, Packages, and Interfaces. |
| **CLOBJ 8** | Demonstrate the Concept of Exception Handling. |
| **CLOBJ 9** | Gain the knowledge of multi-threading. |
| **CLOBJ 10** | Explain the knowledge of Collections Framework. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Explain the fundamental concepts of Object-Oriented Programming (OOP) **and** comprehend the role of Java as an object-oriented and internet-enabled programming language**.** |

| | |
|---|---|
| **CLO 2** | Comprehend and apply Java data types, variables, operators, control statements, and arrays to develop basic programs. |
| **CLO 3** | Develop Java programs using object-oriented concepts such as classes, objects, constructors, methods, inheritance, polymorphism, and packages. |
| **CLO 4** | Implement advanced programming concepts including exception **handling,** strings, interfaces, and multithreading in Java applications. |
| **CLO 5** | Utilize the Java Collection Framework and functional programming **concepts** to design efficient and scalable Java programs. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 2 | 0 | 2 | 3 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage** | **Teaching Hours** |
|---|---|---|---|
| 1 | **Design introduction:** Object-oriented programming, OOPs principles, encapsulation, inheritance, and poly-morphism; Java as an OOP & internet enabled language, importance of Java, Java usage in industry, the byte code, compiling, and running of simple Java program, JVM, JDK, JRE. | 10% | 3 |
| 2 | **Data types, variable, operators:** Data types, variables, dynamic initialization, scope and lifetime of variables, type conversion and casting, operators. | 10% | 3 |

| | | | |
|---|---|---|---|
| 3 | **Control statements:** Conditional Statements, Looping Statements, Jump Statements. | 10% | 3 |
| 4 | **Arrays:** Array, Array values and memory storage Structure, Types of Arrays. | 8% | 3 |
| 5 | **Object-oriented programming:** Classes and objects: concepts of classes and objects, declaring objects, assigning object reference variables, methods, constructors, ac- cess control, garbage collection, usage of static with data and methods, usage of final with data, overloading methods and constructors, parameter passing - call by value, recursion, nested classes. | 18% | 5 |
| 6 | **Inheritance:** Inheritance Basics, member access rules, Usage of super key word, forms of inheritance, Method Overriding, Abstract classes, Dynamic method dispatch, Using final with inheritance. | 8% | 2 |
| 7 | **Strings, Packages and Interfaces:** String handling functions, Packages, Class path, importing packages, differences between classes and interfaces, Implementing & Applying interface, enumerations in Java. | 12% | 4 |
| 8 | **Exception Handling:** Exceptions, Types of Exceptions, Handling of Exceptions. | 8% | 2 |
| 9 | **Multi-Threading:** Thread, Usage of threads, Types of threads, Handling Threads. | 10% | 3 |
| 10 | **Collections Framework:** Functional Programming, Collections, Hierarchy of collections. | 6% | 2 |

**i. Text Books:**

1. Introduction to Java Programming (Comprehensive Version) Daniel Liang; Pearson (TextBook)
2. Core Java Volume-II Fundamentals Horstmann & Cornell; Pearson
3. Complete Reference Java 2 Herbert Schildt; TMH

**j. List of Practical:**

1. Write a Java program to display Hello World message in the console window.
2. Write a Java program to perform arithmetic and bitwise operations in a single source program without object creation.
3. Write a Java program to perform arithmetic and bitwise operations using classes, methods, and objects.
4. Write a Java program to display employee details using the Scanner class.
5. Write a Java program to find solutions of a quadratic equation using the quadratic formula and display a message when there are no real solutions.
6. Write a Java program to generate the Fibonacci sequence using recursive and non-recursive methods.

7. Write a Java program to print prime numbers up to a given integer.
8. Write a Java program to perform matrix multiplication and sorting of names in ascending order.
9. Write Java programs to demonstrate Method Overloading, Constructor Overloading, Method Overriding, Abstract Class, Inheritance, and Interface concepts.
10. Write Java programs to demonstrate Exception Handling including multiple catch blocks, finally block, and user-defined exceptions.
11. Write Java programs to demonstrate Multithreading and Collection Framework including producer–consumer problem, multi-threaded application, and ArrayList operations (add, search, remove)

**Semester 3 - 4**

a. **Course Name:** Digital Electronics

b. **Course Code:** 303105220

c. **Prerequisite:** Basic Electronics

d. **Rationale:** This course is designed to provide basic ideas of computer architec- ture. This course also helps in understanding the organization and architecture of computers and will aid in developing logical abilities.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Identify and explain the digital number system and also be able to justify the practical application of number systems. |
| **CLOBJ 2** | Define and explain different logic gates and codes and how to use them in real-world applications. |
| **CLOBJ 3** | Realize the minimization techniques of digital circuits. |
| **CLOBJ 4** | Design different adders, subtracters, multiplexers, decoders, and many more circuits. |
| **CLOBJ 5** | Apply the theoretical knowledge to design flip-flops, counters, and many more sequential circuits. |
| **CLOBJ 6** | Identify and illustrate specifications of different logic families and memories and analyze them critically. |

**f. Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Learn about digital numbers. |
| **CLO 2** | Know basic principles of all Boolean properties. |
| **CLO 3** | Analyze the logic gates. |
| **CLO 4** | Know how combinational circuits are designed. |
| **CLO 5** | Know how sequential circuits are designed and work. |
| **CLO 6** | Explain the logic families and memory used in computers. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | - | 2 | 3 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage** | **Teaching Hours** |
|---|---|---|---|

| | | | |
|---|---|---|---|
| 1 | **Fundamentals of Digital Systems and Logic Families:** Digital signals, digital circuits, Number Systems: binary, signed binary, octal, hexadecimal numbers, bi- nary arithmetic, one's and two's complements arithmetic, codes, BCD arithmetic, error detecting and correcting codes, AND, OR, NOT, NAND, NOR, and Exclusive-OR operations, examples of IC gates, characteristics of digital ICs, Digital Logic families: TTL and CMOS logic, inter-facing CMOS and TTL. | 15% | 7 |
| 2 | **Minimization Techniques:** Boolean Algebra, Boolean postulates and laws, De-Morgan's Theorem, Principle of Duality, Boolean expression, Minterm, Maxterm, Sum of Products (SOP), Product of Sums (POS), K-map representation, simplification and minimization of logic functions using K-map. Don't care conditions and Quine-McCluskey Method of minimization. Variable Entered Maps, Realizing Logic Function with Gates. | 20% | 8 |
| 3 | **Combinational Digital Circuits:** Binary Adders and Subtractors, Parallel binary adder & subtractor, Serial adder, BCD adder, Carry look-ahead adder, Multiplexer/De-Multiplexer, Encoder/Decoders, Popular MSI chips, Magnitude comparator, parity checker/generator, code converters, priority encoders, de-coders/drivers for display devices. | 20% | 9 |
| 4 | **Sequential Circuits:** A 1-bit memory, the circuit properties of Bi-stable latch, the clocked SR flip-flop, J-K-T and D-type flip-flops, applications of flip-flops, shift registers, Applications of shift registers, ring counter, sequence generator, ripple (Asynchronous) counters, synchronous counters, special counter ICs, asynchronous sequential counters, applications of counters. | 20% | 9 |
| 5 | **A/D and D/A Converters:** Digital to Analog converters: weighted resistor/converter, R-2R Ladder, examples of D to A converters ICs, Analog to Digital converters: successive approximation, A/D converter, dual slope A/D Converter, Example of A/D Converter ICs. | 10% | 5 |
| 6 | **Semiconductor Memories and Programmable Logic Devices:** Classification and characteristics of memories, Content addressable memory (CAM), commonly used memory chips, Introduction to PLD, ROM as a PLD, Programmable Logic Array, Programmable Array Logic, Complex Programmable Logic Devices (CPLDs), Field Programmable Gate Array (FPGA). | 15% | 7 |

**i. Text Book and Reference Books:**

1. Modern Digital Electronics (TextBook) R. P. Jain; Tata McGraw-Hill.
2. Digital Logic and Computer Design Morris Mano; PHI
3. Fundamentals of Digital Circuits Anand Kumar; Prentice-Hall of India Private Limited, New Delhi (2006)

**j. List of Practical:**

1. To study and test various logic gates ICs.
2. Configuring NAND and NOR logic gates as universal gates.
3. Design logic gates using TTL Logic Family.
4. Study and implement Boolean logic functions and combinational circuits like Adder/Subtractor, Code Converters, using logic gates.
5. Study and implement Boolean logic functions and combinational circuits like Multiplexers/De-Multiplexers using logic gates.
6. Study and implement Boolean logic functions and combinational circuits like Encoders/Decoders using logic gates.
7. Study and configure flip-flops using digital ICs. Design digital systems using these circuits.
8. Study and configure registers and counters using digital ICs. Design digital systems using these circuits.
9. Study and design A/D and D/A converters.
10. Introduction to FPGA/CPLD. Implementation of digital circuits studied in previous sessions using PLD/CPLD/FPGA.

**Semester 3 - 5**

**a. Course Name:** Discrete Mathematics

**b. Course Code:** 303191202

**c. Prerequisite:** Knowledge of Mathematics up to 12th science level

**d. Rationale:** The Mathematics I, Mathematics-II syllabus integrates fundamental calculus concepts, advanced mathematical techniques, and vector calculus, preparing students for engineering challenges with optimized problem-solving skills.

**e. Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Apply mathematical techniques to solve diverse real-world problems across different topics in Discrete Mathematics. |
| **CLOBJ 2** | Develop and apply analytical and critical thinking skills to understand, analyze, and evaluate mathematical structures and proofs. |

| | |
|---|---|
| **CLOBJ 3** | Recognize and interpret mathematical solutions within the context of specific problems, demonstrating practical applications in various fields. |
| **CLOBJ 4** | Clearly and effectively communicate mathematical concepts and solutions in both written and verbal forms, adapting to diverse topics. |
| **CLOBJ 5** | Present mathematical arguments and solutions in a unified, logical, and organized manner, emphasizing clarity, coherence, and precision. |
| **CLOBJ 6** | Establish a comprehensive foundation for more advanced courses in mathematics and related disciplines by demonstrating a thorough understanding of fundamental concepts. |

**f. Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Explain logical sentences in terms of predicates, quantifiers, and logical connectives. |
| **CLO 2** | Derive the solution of a given problem using deductive logic |
| **CLO 3** | Classify an algebraic structure of any mathematical problem. |
| **CLO 4** | Evaluate Boolean functions using the properties of Boolean algebra. |
| **CLO 5** | Solve network problems using Graph Theory |
| **CLO 6** | Explain logical sentences in terms of predicates, quantifiers, and logical connectives. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | **Evaluation Scheme** |
|---|---|

| L | T | P | C | Internal Evaluation | | | ESE | | Total |
|---|---|---|---|---|---|---|---|---|---|
| | | | | MSE | CE | P | Theory | P | |
| 4 | - | - | 4 | 20 | 20 | - | 60 | - | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| Sr. No. | Topics | Weightage | Teaching Hours |
|---|---|---|---|
| 1 | **UNIT 1: Sets, Relation and Function**<br>Cartesian Products, Binary Relation, Partial Ordering Relation, Equivalence Relation, Size of a Set, Finite and Infinite Sets, Countable and Uncountable Sets, Can- tor's Diagonal Argument and The Power Set Theorem, Schroeder-Bernstein Theorem. | 11% | 6 |
| 2 | **UNIT 2: Principles of Mathematical Induction**<br>The Well-Ordering Principle, Recursive Definition, The Division Algorithm: Prime Numbers, The Greatest Com- mon Divisor: Euclidean Algorithm, The Fundamental Theorem of Arithmetic. Basic Counting Techniques: Inclusion and Exclusion, Pigeon-Hole Principle, Permutation and Combination. | 9% | 5 |
| 3 | **UNIT 3: Propositional Logic**<br>Syntax, Semantics, Validity and Satisfiability, Basic Connectives and Truth Tables, Logical Equivalence: The Laws of Logic, Logical Implication, Rules of Inference, The Use of Quantifiers. Proof Techniques: Some Terminology, Proof Methods and Strategies, Forward Proof, Proof by Contradiction, Proof by Contraposition, Proof of Necessity and Sufficiency. | 18% | 11 |

| | | | |
|---|---|---|---|
| 4 | **UNIT 4: Algebraic Structures and Morphism**<br>Algebraic Structures with One Binary Operation: Semi Groups, Monoids, Groups, Congruence Relation and Quotient Structures, Free and Cyclic Monoids and Groups, Permutation Groups, Substructures, Normal Subgroups.<br>Algebraic Structures with Two Binary Operations: Rings, Integral Domain and Fields. Boolean Algebra and Boolean Ring, Identities of Boolean Algebra, Duality, Representation of Boolean Function, Disjunctive and Conjunctive Normal Form. | 40% | 24 |
| 5 | **UNIT 5: Graphs and Trees**<br>Graphs and Their Properties: Degree, Connectivity, Path, Cycle, Sub Graph, Isomorphism, Eulerian and Hamiltonian Walks, Graph Colouring, Colouring Maps and Planar Graphs, Colouring Vertices, Colouring Edges, List Colouring, Perfect Graph: Definition, Properties and Example. Rooted Trees, Trees and Sorting, Weighted Trees and Prefix Codes, Bi-connected Component and Articulation Points, Shortest Distances. | 22% | 14 |

**i. Text Books and Reference Books:**


1. Kenneth H. Rosen, Discrete Mathematics and its Applications, Tata McGraw – Hill
2. J.P. Tremblay and R. Manohar, Discrete Mathematical Structure and Its Application to Computer Science, Tata McGraw-Hill
3. Susanna S. Epp, Discrete Mathematics with Applications, 4th Edition, Wadsworth Publishing Co. Inc.
4. C. L. Liu and D. P. Mohapatra, Elements of Discrete Mathematics: A Computer-Oriented Approach, 3rd Edition, Tata McGraw – Hill.

**Semester 3 -6**

a. **Course Name:** Professional Communication Skills

b. **Course Code:** 303193203

c. **Prerequisite:** Knowledge of English language in practical life

d. **Rationale:** Knowledge and application of English, Aptitude and Management Skills are crucial for better employability as well as professionalism.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Demonstrate the ability to communicate clearly and persuasively in oral presentations. |
| **CLOBJ 2** | Practice active listening techniques to enhance understanding in professional interactions. |
| **CLOBJ 3** | Write professional emails, memos, and reports with clarity and conciseness. |
| **CLOBJ 4** | Practice time management strategies effectively. |
| **CLOBJ 5** | Demonstrate skills in resolving conflicts and negotiating effectively. |
| **CLOBJ 6** | Use digital communication tools and platforms effectively. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Apply creative and critical thinking in academic and professional communication tasks. |
| **CLO 2** | Apply vocabulary with accurate pronunciation in academic and professional communication. |
| **CLO 3** | Use fundamental English grammar accurately in academic communication |
| **CLO 4** | Produce effective reading and writing responses for academic, professional, and social contexts. |

| | |
|---|---|
| **CLO 5** | Speak content clearly and appropriately in academic and professional settings, demonstrating growing confidence in their communication skills. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 0 | 2 | 0 | 2 | 0 | 100 | 0 | 0 | 0 | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage** | **Teaching Hours** |
|---|---|---|---|
| 1 | **Technical Writing: Email etiquette & Email writing, Letter Writing (Types of Letters & Layout)** Trains students on detailed email and letter writing etiquette. Students will be able to write formal letters fol- lowing certain stipulated formats. They will learn different types of letters for different official purposes. | 10% | 4 |
| 2 | **Interpersonal Communication at Workplace: Dynamics of communication**<br>To develop the confidence to handle a wide range of demanding situations more effectively at the workplace. To enable the students to analyse their own interpersonal communication style. | 10% | 2 |
| 3 | **Debate: The three-minute debate planner**<br>To enable the students to generate effective critical think- ing into primary issues in the given topic. Students will be able to resolve controversies and recognize strengths and weaknesses of arguments. | 10% | 4 |
| 4 | **Goal setting & Tracking**<br>To enable the students to define strategies or implementation steps to attain the identified goals and make progress every day. | 10% | 2 |

| | | | |
|---|---|---|---|
| 5 | **Time Management & Task Planning (Case-study)**<br>To enable the students to identify their own time wasters and adopt strategies to reduce them. To enable students to clarify and prioritize their objectives and goals by creating more planning time. | 5% | 2 |
| 6 | **Reading Comprehension: Intermediate level**<br>To enable the students to develop the knowledge, skills, and strategies they must possess to become proficient and independent readers. | 5% | 2 |
| 7 | **Listening Skills: Small everyday conversation & comprehension**<br>Provides practice on understanding accents and day-to-day conversations. Listening to English conversations in different contexts. | 10% | 2 |
| 8 | **Information design and writing for print and online media: Blog Writing**<br>To enable students to design information that is targeted to specific audiences in specific situations to meet defined objectives. To create blogs and share their own knowledge and experience with the world. | 5% | 2 |
| 9 | **Advanced vocabulary Building**<br>The students will expand their vocabulary so as to enhance their proficiency in reading and listening to academic texts, writing, and speaking. The students will attain vocabulary to comprehend academic and social reading and listening texts. The students will develop adequate speaking skills to communicate effectively. | 10% | 4 |
| 10 | **Picture Perception**<br>To prepare the students for a test for basic intelligence and IQ, generally done on the first day of SSB (Sashastra Seema Bal is one of India's Central Armed Police Forces). | 5% | 1 |
| 11 | **Appreciation, Apology and Acknowledgement letters**<br>To enable the students to maintain productive business relationships through different types of letters. To enable the students to express their feelings without speaking out loud. | 10% | 2 |
| 12 | **The Art of Negotiation**<br>To enable the students to reach an agreement for mutual benefits through negotiation. To enable the students to learn a process by which compromise or agreement is reached while avoiding argument and dispute. | 5% | 2 |

| | | | |
|---|---|---|---|
| 13 | **Activity Session (Game of Truth)**<br>To make the students think of the significance of certain things in their life. To make them share their thoughts and perceptions of matters in life with others. | 5% | 1 |

**i. Reference Books:**


1. Business Correspondence and Report Writing SHARMA, R. AND MOHAN.
2. Communication Skills Kumar S and Lata P; New Delhi Oxford University
3. Practical English Usage MICHAEL SWAN
4. A Remedial English Grammar for Foreign Students F.T. WOOD
5. On Writing Well William Zinsser; Harper Paperbacks, 2006; 30th anniversary edition


**Semester 4 - 1**

**a. Course Name:** Operating System

**b. Course Code:** 303105251

**c. Prerequisite:** Fundamentals of Computer Systems

**d. Rationale:** This course is an introduction to the theory and practice behind modern computer operating systems. Topics will include what an operating system does (and doesn't) do, system calls and interfaces, processes, concurrent programming, resource scheduling and management, virtual memory, deadlocks, algorithms, programming, and security. The approach of the subject is from both a theoretical perspective as well as a practical one.

**e. Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Gain familiarity with the generation of Operating System, types of operating System, and the concept of a virtual machine. |
| **CLOBJ 2** | Solve problems related to Scheduling Algorithm and concepts of threading, multi-threading, etc. |

| | |
|---|---|
| **CLOBJ 3** | Acquire knowledge of Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution, and Strict Alternation, Peterson's Solution, Semaphores, Event Counters, Monitors, Message Passing, and Classical IPC Problems. |
| **CLOBJ 4** | Define Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection, and Recovery. |
| **CLOBJ 5** | Demonstrate a clear understanding of Memory Management, Memory allocation, and Paging. |
| **CLOBJ 6** | Study Hardware: I/O devices, Device controllers, Direct memory access, Principles of I/O Software: Goals of Interrupt handlers, Device drivers, Device-independent I/O software, etc. |

**f. Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Explain the concepts, types, services, and structures of operating systems. |
| **CLO 2** | Apply process, thread, and scheduling concepts to solve problems related to CPU utilization and turnaround time. |
| **CLO 3** | Evaluate and differentiate IPC and synchronization techniques to resolve classical concurrency problems effectively. |
| **CLO 4** | Apply deadlock prevention and avoidance algorithms, including Banker's algorithm, to solve resource allocation problems. |
| **CLO 5** | Analyse different memory allocation and page replacement strategies to optimize system performance. |
| **CLO 6** | Evaluate and compare I/O, file, and disk management techniques to optimize system performance and reliability. |

**g. Teaching & Examination Scheme:**

| Teaching Scheme | | | | Evaluation Scheme | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| L | T | P | C | Internal Evaluation | | | ESE | | Total |
| | | | | MSE | CE | P | Theory | P | |
| 3 | - | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| Sr. No. | Content | Weightage | Teaching Hours |
|---|---|---|---|
| 1 | **INTRODUCTION:** Concept of Operating Systems, Generations of Operating Systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS- Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine. | 5% | 3 |
| 2 | **PROCESSES, THREAD & PROCESS SCHEDULING:** Processes: Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching. Thread: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads. Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non pre-emptive, FCFS, SJF, RR. | 20% | 9 |
| 3 | **INTER-PROCESS COMMUNICATION:** Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution, Strict Alternation, Peterson's Solution, The Producer/Consumer Problem, Semaphores, Event Counters, Monitors, Message Passing, Classical IPC Problems: Reader's & Writer Problem, Dining Philosopher Problem etc. | 15% | 6 |

| | | | |
|---|---|---|---|
| 4 | **DEADLOCKS:** Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery. | 10% | 5 |
| 5 | **MEMORY MANAGEMENT & VIRTUAL MEM-ORY:** Memory Management: Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation-Fixed and variable partition, Internal and External fragmentation and Compaction; Paging: Principle of operation-Page allocation, Hardware support for paging, Protection and sharing, Disadvantages of paging. Virtual Memory: Basics of Virtual Memory, Hard-ware and control structures, Locality of reference, Page fault, Working Set, Dirty page/Dirty bit, Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Second Chance (SC), Not recently used (NRU) and Least Recently used (LRU). | 30% | 13 |
| 6 | **I/O SYSTEMS, FILE & DISK MANAGEMENT:** I/O Hardware: I/O devices, Device controllers, Direct memory access Principles of I/O Software: Goals of Interrupt handlers, Device drivers, Device independent I/O software. File Management: Concept of File, Ac- cess methods, File types, File operation, Directory struc- ture, File System structure, Allocation methods (contigu- ous, linked, indexed), Free-space management (bit vec- tor, linked list, grouping), directory implementation (lin- ear list, hash table), efficiency and performance. Disk Management: Disk structure, Disk scheduling algorithms - FCFS, SSTF, SCAN, C-SCAN, Disk reliability, Disk for-matting, Boot-block, Bad blocks. | 20% | 9 |

**i. Text Book and Reference Book:**


1. Operating System Concepts Essentials By Avi Silberschatz, Peter Galvin, Greg Gagne — 9th Edition Wiley Asia Student Edition.
2. Operating Systems Internals and Design Principles, By William Stallings — PHI — 5th Edition
3. Operating System: A Design-oriented Approach By Charles Crowley — 1st Edition - Irwin Publishing
4. Operating Systems: A Modern Perspective By Gary J. Nutt — Addison- Wesley — 2nd Edition
5. Design of the Unix Operating Systems By Maurice Bach — Prentice-Hall of India — 8th Edition

6. Understanding the Linux Kernel By Daniel P. Bovet, Marco Cesati — O'Reilly and Associates — 3rd Edition

**j. Practical List**

1. Study of Basic commands of Linux.
2. Study the basics of shell programming.
3. Write a Shell script to print given numbers sum of all digits.
4. Write a shell script to validate the entered date (e.g., Date format: dd-mm-yyyy).
5. Write a shell script to check if the entered string is a palindrome or not.
6. Write a Shell script to say "Good morning", "Good afternoon" , or "Good evening" as you log in to the system.
7. Write a C program to create a child process.
8. Find the largest number from three numbers supplied as command line arguments.
9. Print patterns using a for loop in C.
10. Write a Shell script to determine whether a given file exists or not.
11. Write a C program for process creation using the gcc compiler.
12. Implement the First-Come-First-Served (FCFS) Scheduling Algorithm and Round Robin Scheduling Algorithm.
13. Implement the Banker's Algorithm.

**Semester 4 - 2**

a. **Course Name:** Computer Organization and Microprocessor

b. **Course Code:** 303105210

c. **Prerequisite:** Basic understanding of computer system

d. **Rationale:** This course provides detail of computer system's functional components, their characteristics, performance and interactions including system bus, different types of memory and input/output organization and CPU. This course also covers the architectural issues such as instruction set program and data types. On top that, the students are also introduced to the increasingly important area of parallel organization. This course also serves as a basic to develop hardware- related projects. And hence it is an important course for all students of computer engineering branch.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Explain the fundamental concepts and features of the 8085 microprocessor. |
| **CLOBJ 2** | Describe the architecture of the 8085 microprocessor and interfacing techniques. |
| **CLOBJ 3** | Develop assembly language programs using the instruction set of the 8085 microprocessor. |
| **CLOBJ 4** | Apply advanced programming techniques in 8085 microprocessor-based systems. |
| **CLOBJ 5** | Explain the interrupt structure and interrupt handling mechanisms in the 8085 microprocessor. |
| **CLOBJ 6** | Illustrate computer organization concepts including register transfer operations and basic computer register design. |
| **CLOBJ 7** | Analyse assembler operations and memory organization in computer systems. |

**f. Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Explain the fundamentals of microprocessor 8085, its programming model, instruction set, and data formats. |
| **CLO 2** | Describe the architecture and operations of microprocessors, and demonstrate memory and I/O device interfacing. |
| **CLO 3** | Demonstrate programming methods in 8085 using instructions for looping, counting, indexing, logic operations, rotation, and comparison. |
| **CLO 4** | Implement 8085 programs with counters, delays, stacks, subroutines, and handle interrupts. |
| **CLO 5** | Design register transfer operations, micro-operations, and a basic computer system including instruction codes, control, and interrupt handling. |
| **CLO 6** | Implement assembly programs and memory organization concepts, including cache, virtual, and auxiliary memory. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | - | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage** | **Teaching Hours** |
|---|---|---|---|
| 1 | **UNIT-I: Introduction to Microprocessor 8085**<br>Introduction to Microprocessor 8085, Instruction set and computer languages, 8085 Programming Model, Instruction Data Format and storage. | 10% | 4 |

| | | | |
|---|---|---|---|
| 2 | **UNIT-II: Microprocessor Architecture and Inter- facing**<br>Microprocessor architecture and its operations, Memory and I/O devices, Memory interfacing, Interfacing I/O Devices. | 20% | 8 |
| 3 | **UNIT-III: Programming Methods with Instructions**<br>8085 Instructions, Looping, Counting and Indexing, Logic operations, Rotate and Compare. | 15% | 5 |
| 4 | **UNIT-IV: Additional Programming Techniques**<br>Counter, Time delay, Stack & Subroutines, Restart, Call and Return Instructions, Code conversion. | 15% | 8 |
| 5 | **UNIT-V: 8085 Interrupts**<br>Interrupt structure of 8085 microprocessor, Processing of vectored and non-vectored interrupts, Latency time and response time; Handling multiple interrupts. | 10% | 5 |
| 6 | **UNIT-VI: Computer Organization - Register Transfer and Basic Computer Design**<br>Register Transfer: Register Transfer language, Bus design using multiplexer and Tri-state buffer, Memory Transfers, Arithmetic Micro-Operations, Logic Micro-Operations, Shift Micro-Operations, Arithmetic Logical Shift Unit.<br>Basic Computer Design: Memory-Reference Instructions, Register Reference Instructions, I/O Reference Instructions, Interrupt, Design of Accumulator Unit. | 15% | 8 |
| 7 | **UNIT-VII: Computer Organization - Assembler and Memory Organization**<br>Assembler: Machine Language, Assembly Language, Assembler, Program loops, Programming Arithmetic and Logic operations, Subroutines, I/O Programming. Memory Organization: Memory hierarchy, Main memory, Auxiliary memory, Flash memory, Associative memory, Cache memory, Virtual memory. | 15% | 8 |

**i. Text Books:**


**1.** Microprocessor Architecture, Programming, and Applications with the 8085 (Text Book)
**2.** Computer System Architecture By M.Morris Mano — PHI — 3rd Edition
**3.** Microprocessor 8085 and its Interfacing By Sunil Mathur — PHI Learning Pvt. Ltd
**4.** 8085 Microprocessor And its Applications By A. NagoorKani — TMH Education Pvt. Ltd

**j. List of Practical:**

1. Introduction to Microprocessor 8085 Microprocessor, Instruction set and computer languages, 8085 Programming Model, Instruction Data Format and storage.
2. Microprocessor architecture and interfacing Microprocessor architecture and its operations, Memory and I/O devices, Memory interfacing, Interfacing I/O devices.
3. Programming methods with Instructions 8085 Instructions, Looping, Counting and Indexing, Logic operations Rotate and Compare.
4. Additional Programming techniques Counter, Time delay, Stack Subroutines, Restart, Call and Return Instructions, Code conversion.
5. 8085 Interrupts Interrupt structure of 8085 microprocessor, Processing of vectored and non-vectored interrupts, Latency time and response time; Handling multiple interrupts.
6. Computer Organization - Register Transfer and Basic Computer Design Register Transfer: Register Transfer language, Bus design using multiplexer and Tri state buffer, Memory Transfers, Arithmetic Micro-Operations, Logic Micro- Operations, Shift Micro-Operations, Arithmetic Logical Shift Unit. Basic Computer Design: Instruction codes, Computer registers, Computer instructions, Timing and Control, Instruction cycle, Memory-Reference Instructions, Register Reference Instructions, I/O Reference Instructions, Interrupt, Design of Accumulator Unit.
7. Computer Organization - Assembler and Memory Organization Assembler: Machine Language, Assembly Language, Assembler, Program loops, Programming Arithmetic and Logic operations, Subroutines, I/O Programming. Memory Organization: Memory hierarchy, Main memory, Auxiliary memory, Flash memory, Associative memory, Cache memory, Virtual memory.

**Semester 4 - 3**

a. **Course Name:** Computer Network

b. **Course Code:** 303105255

c. **Prerequisite:** Knowledge of Computer and Information system

d. **Rationale:** This course is designed to provide basic knowledge about data & signals. It also provides basic concepts of computer networks and a firm foundation for understanding how data communication occurs in the Transmission Medium. It will help to develop logical abilities and practically set up the network.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Explain the basic concepts and fundamentals of computer networks. |
| **CLOBJ 2** | Establish basic network connections using appropriate networking tools and configurations. |
| **CLOBJ 3** | Design LAN and WAN network topologies for different communication requirements. |
| **CLOBJ 4** | Describe various switching techniques, routers, and routing tables used in networking |
| **CLOBJ 5** | Demonstrate addressing and address mapping techniques used in computer networks. |
| **CLOBJ 6** | Explain the working and applications of the TCP/IP protocol suite in networking. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Explain data communication components, network topologies, LAN technologies, and bandwidth utilization techniques. |
| **CLO 2** | Apply error control protocols and multiple access techniques to ensure reliable data transmission over networks. |
| **CLO 3** | Analyse network layer mechanisms, including switching, addressing, and routing protocols, to optimize data delivery and forwarding in networks. |
| **CLO 4** | Examine transport layer protocols and congestion control mechanisms to optimize network performance and quality of service. |

| | |
|---|---|
| **CLO 5** | Explain application layer services and protocols including DNS, DDNS, TELNET, EMAIL, FTP, WWW, HTTP, SNMP, Bluetooth, firewalls, and basic cryptography concepts. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | - | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage** | **Teaching Hours** |
|---|---|---|---|
| 1 | **DATA COMMUNICATION COMPONENTS:** Representation of data and its flow Networks, Various Connection Topology, Protocols and Standards, OSI model, Transmission Media, LAN: Wired LAN, Wireless LANs, Connecting LAN and Virtual LAN, Techniques for Bandwidth utilization: Multiplexing - Frequency division, Time division and Wave division, Concepts on spread spectrum | 25% | 11 |
| 2 | **DATA LINK LAYER AND MEDIUM ACCESS SUB LAYER:** Error Detection and Error Correction Fundamentals, Block coding, Hamming Distance, CRC; Flow Control and Error control protocols - Stop and Wait, Go back N ARQ, Selective Repeat ARQ, Sliding Window, Piggybacking, Random Access, Multiple access protocols Pure ALOHA, Slotted ALOHA, CSMA/CD, CDMA/CA. | 25% | 11 |
| 3 | **Network Layer:** Switching, Logical addressing - IPV4, IPV6; Address mapping - ARP, RARP, BOOTP and DHCP Delivery, Forwarding and Unicast Routing protocols | 20% | 8 |

| | | | |
|---|---|---|---|
| 4 | **Transport Layer:** Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), SCTP Congestion Control; Quality of Service, QoS improving techniques: Leaky Bucket and To- ken Bucket algorithm. | 15% | 6 |
| 5 | **Application Layer:** Domain Name Space (DNS), DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls, Basic concepts of Cryptography. | 15% | 6 |

**i. Text Books and Reference Books:**

1. Computer Networks (Text Book) by Andrew S. Tanenbaum and David J. Wetherall — PEARSON Edition
2. Internetworking with TCP/IP Principles, Protocols, and Architecture by Douglas E. Comer
3. Basic Electrical Engineering, By D. C. Kulshreshtha, McGraw Hill, Pub. Year 2009
4. Electrical and Electronics Technology, By E. Hughes Pearson, Pub. Year 2010

**j. List of Practical:**

1. Experiments on Simulation Tools: (CISCO PACKET TRACER).
2. Experiments of Packet capture tool: Wireshark.
3. To study the behaviour of generic devices used for networking: (CISCO PACKET TRACER).
4. Data Link Layer (Error Correction).
5. Virtual LAN.
6. Wireless LAN.
7. Inter networking with routers: 1: Experiment on same subnet 2: Perform Experiment across the subnet and observe functioning of Router via selecting suitable pair of Source and destination.
8. Implementation of SUBNETTING.
9. Routing at Network Layer.
10. Experiment on Transport Layer.

**Semester 4 - 4**

a. **Course Name:** Programming in Python with Full Stack Development

b. **Course Code:** 303105257

c. **Prerequisite:** Basic knowledge of Programming and web applications

d. **Rationale:** This course provides a broad introduction to Python programming and development of web applications. Developing and using Python as a scripting language for automating tasks and data processing. Moreover, building and deploying web applications using popular Python frameworks such as Django and Flask.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Explain the fundamental concepts of web development and basic Python programming. |
| **CLOBJ 2** | Demonstrate the use of functions and object-oriented programming (OOP) concepts in Python. |
| **CLOBJ 3** | Apply modules and packages to organize Python programs efficiently. |
| **CLOBJ 4** | Explain and implement web applications using the Flask framework. |
| **CLOBJ 5** | Develop web applications using the Django framework. |
| **CLOBJ 6** | Explain and utilize RESTful APIs for web service communication. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Explain Python basics, control structures, and core data structures. |
| **CLO 2** | Utilize Python standard libraries, regular expressions, and user-defined functions, modules, and packages. |
| **CLO 3** | Perform database operations, file handling, and manage errors and exceptions in Python programs. |

| | |
|---|---|
| **CLO 4** | Apply object-oriented programming concepts in Python, including classes, objects, methods, inheritance, and data encapsulation. |
| **CLO 5** | Develop graphical and GUI applications in Python using Turtle, Tkinter, and visualize data using NumPy, Pandas, and Matplotlib. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | - | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage** | **Teaching Hours** |
|---|---|---|---|
| 1 | **UNIT-I: Introduction to Python Programming**<br>Introduction to Python and basic programming concepts, variables, data types, conditional statements and loops, Lists, Sets, Tuples, Dictionaries: Working with strings, lists, sets, tuples and dictionaries, including common operations and built-in functions. | 15% | 6 |
| 2 | **UNIT-II: Functions and OOPS Concepts**<br>Defining and using functions, including the use of arguments and return values. OOPS Concepts: Object, class, abstraction, encapsulation, polymorphism, Inheritance. Exceptions and File handling: Handling exceptions and working with files. | 20% | 5 |

| | | | |
|---|---|---|---|
| 3 | **UNIT-III: Modules and Packages**<br>Working with modules and packages in Python. Introduction to popular Python libraries for specific tasks, such as data analysis, web development, or game development. PyCharm IDE: GIT- Git Integration with PyCharm IDE, PyTests. Python connectivity with Databases MYSQL, MongoDB CRUD operations. | 15% | 5 |
| 4 | **UNIT-IV: Flask Framework**<br>Introduction to Flask and web development with Python, Installation in Virtual Environment. Creation Routing App Settings URL Building HTTP methods Templates Working with Static, Media Files. Sending Form Data to Template. Flask App with Database connectivity Sqlite3, MySQL. Handling Exceptions and Errors Flash Message Working with Mails. Authenticating and authorizing users with Flask-Login, Deploying a Flask application to a web server. | 20% | 10 |
| 5 | **UNIT-V: Django Framework**<br>Introduction to Django framework, Django Project Installation in Virtual Environment. Phases in Django Project Creation Create a Project. Creation of Apps and their Structure. Working with ADMIN Console. Creating Views URL Mapping. Template System Working with Models. Form Processing static, media files, Django App Deployment. | 20% | 10 |
| 6 | **UNIT-VI: RESTful APIs**<br>Introduction to RESTful APIs and the REST architectural style, Understanding the HTTP protocol and its role in RESTful APIs, Designing and implementing RESTful APIs using common HTTP methods, such as GET, POST, PUT, and DELETE, Using URLs and resource representations to identify and transfer data in RESTful APIs, Implementing best practices for designing and implementing RESTful APIs, such as using HTTP status codes, versioning, and error handling, Consuming RESTful APIs using common tools and libraries, such as cURL, Postman, and the requests library in Python, Building scalable and secure RESTful APIs using common frame- works and libraries Flask or FastAPI. | 10% | 6 |

**i. Text Books and Reference Books:**


1. Fluent Python, 2nd Edition by Luciano Ramalho
2. Learn Python3 the Hard Way By Zed Shaw
3. Django for Beginners: Build websites with Python and Django by William S. Vincent.

4. Learning Django Web Development by Samuli Natri.
5. Flask Web Development with Python by Miguel Grinberg.
6. Mastering Flask by Jack Stouffer.
7. Building RESTful Python Web Services by Gasto´n C. Hillar.
8. Building Web APIs with FastAPI by Samuel Colvin.

**j. List of Practical:**

**1. Set-1**

1. A program that converts temperatures from Fahrenheit to Celsius and vice versa.
2. A program that calculates the area and perimeter of a rectangle.
3. A program that generates a random password of a specified length.
4. A program that calculates the average of a list of numbers.
5. A program that checks if a given year is a leap year.
6. A program that calculates the factorial of a number.
7. A program that checks if a given string is a palindrome.
8. A program that sorts a list of numbers in ascending or descending order.
9. A program that generates a multiplication table for a given number.
10. A program that converts a given number from one base to another.

**2. Set-2**

1. A program that models a bank account, with classes for the account, the customer, and the bank.
2. A program that simulates a school management system, with classes for the students, the teachers, and the courses.
3. A program that reads a text file and counts the number of words in it.
4. A program that reads a CSV file and calculates the average of the values in a specified column.
5. A program that reads an Excel file and prints the data in a tabular format.

**3. Set-3**

1. A program that creates a simple web server and serves a static HTML page.
2. A program that creates a web application that allows users to register and login.
3. A program that creates a web application that allows users to upload and download files.
4. A program that creates a web application that displays data from a database in a tabular format.
5. A program that creates a web application that accepts user input and sends it to a server-side script for processing.

4. **Set-4**
    1. A program that creates a web application that uses a template engine to generate dynamic HTML pages.
    2. A program that creates a web application that supports AJAX requests and updates the page without reloading.
    3. A program that creates a web application that uses Django's built-in debugging features to troubleshoot errors and exceptions.
    4. A program that creates a web application that implements user authentication and authorization.
    5. A program that creates a web application that integrates with third-party APIs to provide additional functionality.
5. **Set-5**
    1. A program that creates a simple RESTful API that returns a list of users in JSON format.
    2. A program that creates a RESTful API that allows users to create, read, update, and delete resources.
    3. A program that creates a RESTful API that authenticates users using a JSON Web Token.
    4. A program that creates a RESTful API that paginates the results of a query to improve performance.
    5. A program that creates a RESTful API that supports data validation and error handling.

**Semester 4-5**

a. **Course Name:** Probability Statistics and Numerical Method

b. **Course Code:** 303191258

c. **Prerequisite:** Knowledge of Mathematics up to 12th science level

d. **Rationale:** The Mathematics I, Mathematics-II syllabus integrates fundamental calculus concepts, advanced mathematical techniques, and vector calculus, preparing students for engineering challenges with optimized problem-solving skills.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Define fundamental concepts of probability, probability spaces, conditional probability, and Bayes' Rule for making informed statistical decisions. |
| **CLOBJ 2** | Analyse discrete and continuous random variables, compute expectations and variances, and explore key distributions such as Binomial, Poisson, and Normal. |
| **CLOBJ 3** | Develop skills in hypothesis testing, including large sample tests for proportions , means, standard deviations, and chi-square tests for goodness of fit and independence. |
| **CLOBJ 4** | Demonstrate proficiency in numerical methods for solving linear equations, finding roots of algebraic and transcendental equations using techniques like Gauss-Jacobi, Gauss-Seidel, Bisection, Newton-Raphson, and Regula Falsi. |
| **CLOBJ 5** | Master the concepts of finite differences, interpolation using Newton's Forward and Backward Difference Formula, Newton's Divided, and Lagrange's Formula for Unequal Intervals. |
| **CLOBJ 6** | Gain expertise in numerical integration techniques such as the Trapezoidal rule, Simpson's 1/3rd and 3/8th Rules, Gaussian Quadrature, and solve ordinary differential equations using methods like Taylor's series, Euler, Modified Euler, and Runge Kutta of the fourth order for first and second-order equations. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Calculate and interpret correlation and regression for two variables.. |
| **CLO 2** | Formulate and solve problems involving discrete and continuous random variables. |
| **CLO 3** | Apply appropriate statistical methods to analyse and interpret experimental data. |
| **CLO 4** | Design and implement numerical techniques for interpolation and the solution of differential equations. |
| **CLO 5** | Explain the significance of numerical methods in solving real-world problems where analytical methods are not feasible. |
| **CLO 6** | Compare and evaluate numerical integration methods for solving different types of problems. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 4 | - | - | 4 | 20 | 20 | - | 60 | - | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation;

**CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| Sr. No. | Topics | Weightage | Teaching Hours |
|---|---|---|---|
| 1 | **UNIT 1: Correlation, Regression and Curve Fit- ting** Correlation and Regression – Rank correlation, Curve Fit-ting by The Method of Least Squares- Fitting of Straight Lines, Second Degree Parabolas and More General Curves. | 18% | 8 |
| 2 | **UNIT 2: Probability and Probability Distributions** Probability Spaces, Conditional Probability, Bayes' Rule, Discrete and Continuous Random Variables, Independent Random Variables, Expectation and Variance of Discrete and Continuous Random Variables, Distribution and Their Properties: Binomial Distribution, Poisson Distribution, Normal Distribution. | 23% | 10 |
| 3 | **UNIT 3: Testing of Hypothesis** Test of significance: Large sample test for single proportion, difference of proportions, single mean, difference of means, and difference of standard deviations. Test for single mean, difference of means, Test for ratio of variances, Chi-square test for goodness of fit and independence of attributes. | 25% | 15 |
| 4 | **UNIT 4: Solution of a System of Linear Equations, Roots of Algebraic and Transcendental Equations** Gauss-Jacobi and Gauss-Seidel Methods, Solution of Polynomial and Transcendental Equations: Bisection Method, Newton-Raphson Method and Regula Falsi Method. | 18% | 10 |
| 5 | **UNIT 5: Finite Differences and Interpolation** Finite Differences, Relation between Operators, Interpolation using Newton's Forward and Backward Difference Formula. Newton's Divided and Lagrange's Formula for Unequal Intervals. | 10% | 6 |
| 6 | **UNIT 6: Numerical Integration** Trapezoidal rule, Simpson's 1/3rd and 3/8th Rules, Gaussian Quadrature Formulae. | 6% | 2 |

**i. Text Books:**


1. B. S. Grewal, "Numerical Methods in Engineering & Science with Programs in C and C++", Khanna Publishers.
2. C.E. Froberg, "Introduction to Numerical Analysis", Addison-Wesley.

**3.** P. G. Hoel, S. C. Port and C. J. Stone, “Introduction to Probability Theory”, Universal Book Stall, 2003 (Reprint).

**4.** S.C. Gupta and V. K. Kapoor, “Fundamentals of Mathematical Statistics”, Sultan Chand & Sons

**Semester 4 - 6**

a. **Course Name:** Professional Grooming & Personality Development

b. **Course Code:** 303193252

c. **Prerequisite:** Knowledge of English language in practical life

d. **Rationale:** Knowledge and application of English, Aptitude, and Management Skills are crucial for better employability as well as professionalism.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Demonstrate the ability to communicate clearly and effectively in oral presentations. |
| **CLOBJ 2** | Practice active listening techniques to improve understanding in professional interactions. |
| **CLOBJ 3** | Prepare professional emails, memos, and reports with clarity, accuracy, and conciseness. |
| **CLOBJ 4** | Apply effective time management strategies in academic and professional activities. |
| **CLOBJ 5** | Demonstrate conflict resolution and negotiation skills in professional situations. |
| **CLOBJ 6** | Use digital communication tools and platforms effectively for professional communication. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Cultivate essential soft skills for personal effectiveness and professional growth in both academic and workplace settings. |
| **CLO 2** | Practice professional etiquette and collaborative behaviour in organizational environments |
| **CLO 3** | Participate effectively in oral organizational communication through clear speaking, active listening, and appropriate interaction in group settings. |

| | |
|---|---|
| **CLO 4** | Analyse texts using appropriate reading comprehension strategies |
| **CLO 5** | Develop assertive and professional ideas for effective communication in academic and workplace environments |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 0 | 1 | 0 | 1 | 0 | 100 | 0 | 0 | 0 | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage** | **Teaching Hours** |
|---|---|---|---|
| 1 | **Self Development and Assessment:** Various self-assessments for personal and professional development skills that are relevant to career development: Change, Grow, Persist, Prioritize, Read, Learn, Listen, Record, Remember, Guess, Think, Communicate, Relate, and Dream. | 25% | 4 |
| 2 | **Corporate Etiquette:** Tips and guide to develop personality and gain various etiquettes manners, case studies, and activities. Telephone etiquette Etiquette for foreign business trips Etiquette for small talks Respecting privacy Learning to say 'No' | 25% | 4 |

| | | | |
|---|---|---|---|
| 3 | **Public Speaking:** It's process of communicating information to an audience and is helpful in career advancement. Effective Public speaking skills includes: Choosing appropriate pattern Selecting appropriate method Art of persuasion Making speeches effective Delivering different types of speeches | 20% | 4 |
| 4 | **Reading Skills Activity & Reading Comprehension:** Aims to improve students' comprehensive skills in English Language by getting them involved in reading activity and providing practice for reading comprehension. | 15% | 2 |
| 5 | **Listening Skills- Inquiry Based Listening Questions:** Aims to improve students' listening skills in English Language providing them practice of various types of inquiry based listening tracks. Students will listen and will be able to find out details from the conversations. | 15% | 1 |

*Continuous Evaluation: It consists of Assignments/Seminars/Presentations/Quizzes/ Surprise Tests (Summative/MCQ) etc.

**i. Reference Books:**


1. Business Correspondence and Report Writing SHARMA, R. AND MOHAN, K.
2. Communication Skills Kumar S and Lata P; New Delhi Oxford University Press
3. Practical English Usage MICHAEL SWAN
4. A Remedial English Grammar for Foreign Student F.T. WOOD

**Semester 4 - 7**

a. **Course Name:** Competitive Coding

b. **Course Code:** 303105259

c. **Prerequisite:** Computer Programming and Basic Syntaxes

d. **Rationale:** Competitive coding enhances problem-solving abilities by exposing individuals to a wide range of algorithmic challenges. It fosters critical thinking and quick decision-making skills, crucial for real-world problem-solving in computer science and software development. Engaging in competitive coding also promotes continuous learning and improvement, as participants strive to optimize solutions and compete against global coding communities.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Analyse and assess time complexity in algorithmic solutions, enabling the ability to make informed judgments during problem-solving processes. |
| **CLOBJ 2** | Demonstrate proficiency in the application of various sorting algorithms, employing them effectively to organize and manipulate data structures for problem-solving purposes. |
| **CLOBJ 3** | Evaluate problem requirements and make informed decisions on selecting the most suitable data structure to optimize solution efficiency and address specific problem constraints. |
| **CLOBJ 4** | Employ diverse problem-solving techniques to effectively tackle a range of challenges, showcasing adaptability and resourcefulness in approaching and resolving different types of problems. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Judge time complexity rules during problem solving. |
| **CLO 2** | Apply sorting algorithms to data structures to solve problems. |
| **CLO 3** | Select the best data structure to solve the given problem. |

| | |
|---|---|
| **CLO 4** | Solve given problems using different Problem-Solving Techniques. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **T** | **P** | **CE** | **Theory** | **P** | |
| - | - | 4 | 2 | - | 20 | - | - | 30 | 50 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. List of Practical:**

1. Write a program for implementing a MINSTACK which should support operations like push, pop, overflow, underflow, display.
   i. Construct a stack of N-capacity
   ii. Push elements
   iii. Pop elements iv. Top element
   iv. Retrieve the min element from the stack
2. Write a program to deal with real-world situations where Stack data structure is widely used. Evaluation of expression: Stacks are used to evaluate expressions, especially in languages that use postfix or prefix notation. Operators and operands are pushed onto the stack, and operations are performed based on the LIFO principle.
3. Write a program for finding the Next Greater Element (NGE) from an array.
4. Write a program to design a circular queue (k) which should implement the following functions:
   i. Enqueue
   ii. Dequeue
   iii. Front
   iv. Rear
5. Write a program for an infix expression, and convert it to postfix notation. Use a queue to implement the Shunting Yard Algorithm for expression conversion.
6. Write a program for finding the Product of the three largest Distinct Elements. Use a Priority Queue to efficiently find and remove the largest elements.

7. Write a program to Merge two sorted linked lists.
8. Write a program to find the Merge point of two sorted linked lists.
9. Write a program to Swap Nodes pairwise in a linked list.
10. Write a program for building a function ISVALID to validate a Binary Search Tree (BST).
11. Write a program to Build a BST.
12. Write a program to determine the depth of a given Tree by implementing MAXDEPTH.
13. Write a program to understand and implement Tree traversals, i.e., Pre-Order, Post-Order, In-Order.
14. Write a program to perform Boundary Traversal on a BST.
15. Write a program for Lowest Common Ancestors (LCA) in a BST.
16. Write a program to verify and validate mirrored trees.
17. Write a program for a basic hash function in a programming language of your choice. Demonstrate its usage to store and retrieve key-value pairs.
18. Implement a hash table using separate chaining for collision handling. Perform operations like insertion, deletion, and search on the hash table.
19. Write a program to implement Two Sums using a HashMap.
20. Write a program to implement search, insert, and remove operations in a Trie.
21. Write a program to implement Huffman coding.
22. Write a program to find distinct substrings in a string.
23. Write a program to find the number of words in a Trie.
24. Write a program to view a tree from the left view.
25. Write a program to traverse a tree using Level Order Traversal.

**i. Text Books:**


1. Robert Sedgewick and Kevin Wayne, Algorithms, Part I and II, Addison- Wesley.
2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, Introduction to Algorithms, MIT Press.
3. Steven S. Skiena, The Algorithm Design Manual, Springer.
4. Mark Allen Weiss, Data Structures and Algorithm Analysis in C++, Addison-Wesley.
5. Sahni, A. and Horowitz, E., Fundamental Data Structures, McGraw-Hill.

SEMESTER -5

**Semester 5-1**

a. **Course Name:** Design and Analysis of Algorithm

b. **Course Code:** 303105218

c. **Prerequisite:** Data structures, Fundamentals of programming

d. **Rationale:** Analyze the asymptotic performance of algorithms. Write rigorous correctness proofs for algorithms. Demonstrate a familiarity with major algorithms and data structures. Apply important algorithmic design paradigms and methods of analysis. Synthesize efficient algorithms in common engineering design situations.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Develop the ability to analyze the running time of any given algorithm using asymptotic analysis and prove the correctness of basic algorithms. |
| **CLOBJ 2** | Design efficient algorithms for computational problems, using various algorithm design techniques taught in the course. |
| **CLOBJ 3** | Explain the major graph algorithms and their analyses. Employ graphs to model engineering problems, when appropriate. |
| **CLOBJ 4** | Analyze different string matching algorithms based on their methodologies and time–space complexities. |
| **CLOBJ 5** | Explain the complexity classes P, NP, and NP-Complete, and demonstrate the NP-Completeness of a specific problem. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Analyze algorithm characteristics and complexities using asymptotic notations and recurrence relations. |

| | |
|---|---|
| **CLO 2** | Apply fundamental algorithmic strategies (brute-force, greedy, DP, backtracking, branch-and-bound, heuristics) to solve computational problems. |
| **CLO 3** | Implement graph and tree algorithms for traversal, shortest path, spanning tree, and network flow problems. |
| **CLO 4** | Differentiate tractable and intractable problems, and classify algorithms into P, NP, NP-complete, and NP-hard categories. |
| **CLO 5** | Evaluate approximation, randomized algorithms, and PSPACE problems. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | - | 4 | 5 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage (%)** | **Teaching Hours** |
|---|---|---|---|

| | | | |
|---|---|---|---|
| 1 | Introduction and Analysis of Algorithms:<br>Algorithm: Definition, Properties, Types of Algorithms, Writing an Algorithm, Analysis: Parameters, Design Techniques of Algorithms<br>Asymptotic Analysis: Big Oh, Big Omega & Big Theta Notations, Lower Bound, Upper Bound and Tight Bound, Best Case, Worst Case, Average Case, Analyzing control statement, Loop invariant and the correctness of the algorithm, Recurrences-substitution method, recursion tree method, master method. Sorting Techniques with analysis: Bubble Sort, Selection Sort, Insertion Sort. | 20% | 10 |
| 2 | Divide & Conquer Algorithms: Structure of divide-and conquer algorithms, Examples: Binary search, Quick sort, Merge sort, Strassen's Multiplication, Max-Min problem. | 20% | 6 |
| 3 | Greedy Algorithms: Introduction, Elements of Greedy Strategy, Minimum Spanning Tree: Kruskal's & Prim's Algorithm, Dijkstra's Algorithm, Knapsack Problem, Activity Selection Problem, Huffman Codes. | 20% | 8 |
| 4 | Dynamic Programming: Principle of Optimality, 0/1 Knapsack Problem, Making Change Problem, Chain Matrix Multiplication, Longest Common Subsequence, All Pair Shortest Paths: Warshall's and Floyd's Algorithms. | 20% | 8 |
| 5 | Exploring Graphs: An introduction using graphs and games, Undirected Graph, Directed Graph, Traversing Graphs, Depth First Search, Breadth First Search, Topological Sort. | 5% | 3 |
| 6 | Backtracking and Branch & Bound: Introduction to Backtracking, Introduction to Branch & Bound, 0/1 Knapsack Problem, N-Queens Problem, Travelling Salesman Problem. | 5% | 4 |
| 7 | String Matching & NP Completeness: String Matching: Introduction to String Matching, Naive String Matching, Rabin-Karp Algorithm, Kruth-Morris-Pratt Algorithm, String Matching using Finite Automata. NP Completeness: Introduction to NP Completeness, P Class Problems, NP Class Problems, Hamiltonian Cycle. | 10% | 6 |

**i. Reference Books:**


1. Introduction to Algorithms, 4TH Edition by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, MIT Press/McGraw-Hill. (TextBook)
2. Fundamentals of Algorithms by E. Horowitz et al. (Text Book)
3. Algorithm Design, 1ST Edition by Jon Kleinberg and Eva Tardos, Pearson.´

4. Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition by Michael T. Goodrich and Roberto Tamassia, Wiley.
5. Algorithms—A Creative Approach, 3RD Edition" by Udi Manber, Addison Wesley, Reading, MA.

**j. List of Practical:**

1. Write a program to determine whether the given number is Prime or not.
2. Given a sorted array and a target value, return the index if the target is found. If not, return the index where it would be if it were inserted in order.
3. There are N children standing in a line with some rating value. You want to distribute a minimum number of candies to these children such that:
   - Each child must have at least one candy.
   - The children with higher ratings will have more candies than their neighbours.

   You need to write a program to calculate the minimum candies you must give.
4. There is a new barn with N stalls and C cows. The stalls are located on a straight line at positions $x_1,x_N$ ($0 \le x_i \le 1{,}000{,}000{,}000$). We want to assign the cows to the stalls, such that the minimum distance between any two of them is as large as possible. What is the largest minimum distance?
5. Given an undirected graph with V vertices and E edges, check whether it contains any cycle or not.
6. There are n servers numbered from 0 to n−1 connected by undirected server-to server connections forming a network where connections[$i$] = [$a_i$,$b_i$] represents a connection between servers $a_i$ and $b_i$. Any server can reach other servers directly or indirectly through the network. A critical connection is a connection that, if removed, will make some servers unable to reach some other servers. Return all critical connections in the network in any order.
7. Given a grid of size $N \times M$ (N is the number of rows and M is the number of columns in the grid) consisting of '0's (Water) and '1's (Land). Find the number of islands.
8. Given a grid of dimension $N \times M$ where each cell in the grid can have values 0, 1, or 2 which has the following meaning:
   - 0: Empty cell
   - 1: Cells have fresh oranges
   - 2: Cells have rotten oranges

   We have to determine what is the minimum time required to rot all oranges. A rotten orange at index [$i$,$j$] can rot other fresh oranges at indexes [$i-1$,$j$], [$i+1$,$j$], [$i$,$j-1$], [$i$,$j+1$] (up, down, left and right) in unit time.
9. Given two strings str1 and str2 and below operations that can be performed on str1. Find the minimum number of edits (operations) required to convert

'str1' into 'str2'.

- Insert
- Remove • Replace

All of the above operations are of equal cost.

10. The "Minimum Path Sum" problem states that given an $n \times m$ grid consisting of non-negative integers, we need to find a path from top-left to bottom-right, which minimizes the sum of all numbers along the path.
11. Given string num representing a non-negative integer num, and an integer k, return the smallest possible integer after removing k digits from num.
12. There is a robot on an $m \times n$ grid. The robot is initially located at the top-left corner (i.e., grid[0][0]). The robot tries to move to the bottom-right corner (i.e., grid[$m-1$][$n-1$]). The robot can only move either down or right at any point in time. Given the two integers m and n, return the number of possible unique paths that the robot can take to reach the bottom-right corner.

**Semester 5-2**

a. **Course:** Theory of Computation

b. **Course Code:** 303105306

c. **Prerequisite:** Calculus, Data Structures, and Algorithms

d. **Rationale:** Formal Language & Automata Theory helps in natural language processing to solve a problem on a model of computation, using an algorithm. It enables to learn in which machine can be made to think.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLO 1** | Define the basic concepts of languages, grammars, and computation models. |
| **CLO 2** | Apply finite automata and regular expressions to solve computational problems. |
| **CLO 3** | Utilize context-free grammars and pushdown automata in language design and processing. |
| **CLO 4** | Analyze the capabilities and limitations of Turing Machines and recursive languages. |
| **CLO 5** | Explore undecidability and theoretical foundations of computation. |

f. **Course Outcome:**

| | |
|---|---|
| **CO 1** | Recognize the basic concepts and applications of theory of Computation. |
| **CO 2** | Solve Computational Problems using Regular Languages and Finite Automata. |
| **CO 3** | Solve Computational Problems using Context free Grammar and Push Down Automata. |
| **CO 4** | Design Turing Machine for simple computational Problems. |
| **CO 5** | Analyze various concepts of undecidability and Computable Function. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | 0 | 0 | 3 | 20 | 20 | - | 60 | - | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage (%)** | **Teaching Hours** |
|---|---|---|---|
| 1 | **Introduction:**<br>Alphabet, languages and grammars, productions and derivation, Chomsky hierarchy of languages | 5% | 2 |
| 2 | **Regular languages and finite automata:**<br>Regular expressions and languages, DFA and equivalence with REs, Moore & Mealy machines, conversion between Mealy and Moore, NFA and equivalence with DFA, regular grammars, pumping lemma, minimization of finite automata. | 30% | 12 |
| 3 | **Grammars:**<br>Context-free grammars (CFG), Chomsky normal forms, PDA and CFG equivalence, ambiguity, pumping lemma for CFLs, deterministic PDA, context-sensitive grammars and languages. | 35% | 15 |

| | | | |
|---|---|---|---|
| 4 | **Turing machines:**<br>Basic TM model, Turing-recognizable and decidable languages, variants, nondeterministic TM, unrestricted grammars, TMs as enumerators. | 25% | 10 |
| 5 | **Undecidability:**<br>Church-Turing thesis, universal TM, diagonalization languages. | 5% | 6 |

**i. Reference Books:**


1. *Introduction to Automata Theory, Languages and Computation* by John E. Hopcroft, Rajiv Motwani, Jeffrey D. Ullman – Pearson (TextBook)
2. *Elements of the Theory of Computation* by Harry R. Lewis and Christos H. Papadimitriou – Pearson Education Asia
3. *Introduction to the Theory of Computation* by Michael Sipser – PWS Publishing
4. *Introduction to Languages and the Theory of Computation* by John C. Martin – McGraw Hill
5. *Automata and Computability* by Dexter C. Kozen – Springer


**Semester 5 - 3**

**a. Course Name:** Software Engineering

**b. Course Code:** 303105253

**c. Prerequisite:** Basic knowledge of software applications

**d. Rationale:** This course provides a broad introduction to software engineering. The various process models required to develop software are also described. Moreover, the functional and non-functional requirements are also described.

**e. Course Learning Objectives:**

| | |
|---|---|
| **CLO1** | Define software and explain its role, components, and importance in computer systems. |
| **CLO2** | Create simple software applications by applying basic programming concepts and development tools. |
| **CLO 3** | Design a basic Wide Area Network (WAN) or Local Area Network (LAN) architecture illustrating its components, topology, and connectivity. |
| **CLO4** | Explain different types of software, including system software, application software, and utility software, along with their functions. |
| **CLO5** | Demonstrate the use of software tools and techniques for testing applications to ensure functionality and reliability. |

| | |
|---|---|
| **CLO6** | Study about the Use Case study, CASE Tools, and Advanced Practices of System Dependability and Security. |

**f. Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Explain software process models and apply project management principles for planning, scheduling, and risk management. |
| **CLO 2** | Analyze software requirements and design systems using architectural, component, and data-oriented approaches. |
| **CLO 3** | Apply coding standards, programming principles, and unit testing techniques to develop, evaluate, and measure software quality using metrics. |
| **CLO 4** | Analyze software testing techniques, quality assurance processes, and standards to ensure reliability and effectiveness of software systems. |
| **CLO 5** | Utilize CASE tools and advanced practices for software dependability, security, and agile development. |
| **CLO 6** | Examine advanced software engineering paradigms, including component-based, distributed, and service-oriented systems. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | - | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Content** | **Weightage** | **Teaching Hours** |
|---|---|---|---|

| | | | |
|---|---|---|---|
| 1 | **Introduction:** Study of Different Models, Software Characteristics, Components, Applications, Layered Technologies, Processes, Methods and Tools, Generic View Of Software Engineering, Process Models - Waterfall model, Incremental, Evolutionary process models - Prototype, Spiral And Concurrent Development Model; **Agile Development:** Agility and Agile Process model, Extreme Programming, Other process models of Agile Development and Tools. | 10% | 6 |
| 2 | **Software Project Management:** Management Spectrum, People – Product – Process – Project, W5HH Principle, Importance of Team Management; Planning a Software Project: Scope and Feasibility, Effort Estimation, Schedule and Staffing, Quality Planning, Risk Management - Identification, Assessment, Control, Project Monitoring Plan, Detailed Scheduling. | 10% | 5 |
| 3 | **Requirements Engineering:** Problem Recognition, Requirement Engineering Tasks, Processes, Requirements Specification, Use Cases and Functional Specification, Requirements Validation, Requirements Analysis. | 10% | 5 |
| 4 | **Structured System Design:** Design Concepts, Design Model, Software Architecture, Data Design, Architectural Styles and Patterns, Architectural Design, Alternative Architectural Designs, Modelling Component Level Design and Its Modelling, Procedural Design, Object-Oriented Design. **Data Oriented Analysis & Design:** Difference between Data and Information, E-R Diagram, Dataflow Model, Control Flow Model, Control and Process Specification, Data Dictionary. | 15% | 5 |
| 5 | **Coding and Unit Testing:** Programming Principles and Guidelines, Programming Practices, Coding Standards, Incremental Development of Code, Management of Code Evaluation, Unit Testing - Procedural Units, Classes, Code Inspection, Metrics - Size Measure, Complexity Metrics, Cyclomatic Complexity, Halstead Measure, Knot Count, Comparison of Different Metrics. | 10% | 4 |
| 6 | **Software Testing and Quality Assurance:** Concepts, Psychology of Testing, Levels of Testing, Testing Process - Test Plan, Test Case Design, Execution, Black-Box Testing – Boundary Value Analysis – Pairwise Testing – State Based Testing, White-Box Testing Criteria and Test Case Generation and Tool Support; **Quality Assurance:** Quality Control, Assurance, Cost, Reviews, Software Quality Assurance, Approaches to SQA, Reliability, Quality Standards - ISO9000 and 9001. | 15% | 7 |

| | | | |
|---|---|---|---|
| 7 | **CASE Tools and Advanced Practices of System Dependability and Security:** Computer Aided Software Engineering Tools, SCRUM Developments, Dependable System, Reliability Engineering, Safety Engineering, Security Engineering, Resilience Engineering. | 15% | 5 |
| 8 | **Advanced Software Engineering:** Software Reuse, Component Based Software Engineering, Distributed Software Engineering, Service-Oriented Software Engineering, Real-Time Software Engineering, Systems Engineering, Systems of System. | 15% | 5 |

**i. Text Books and Reference Books:**

1. Software Engineering (Text Book) R. Pressman; 6th Edition
2. Internetworking with TCP/IP Principles, Protocols and Architecture
3. Software Engineering By Sommerville
4. Data Communication and Networking

**j. List of Practicals:**

1. Project Definition and objective of the specified module and Perform Requirement Engineering Process.
2. Identify Suitable Design and Implementation model from the different software engineering models.
3. Prepare Software Requirement Specification (SRS) for the selected module.
4. Develop Software Project Management Planning (SPMP) for the specified module.
5. Do Cost and Effort Estimation using different Software Cost Estimation models.
6. Prepare System Analysis and System Design of identified Requirement Specification using structure design as DFD with data dictionary and Structure Chart for the specific module.
7. Designing the module using Object-Oriented approach including Use Case Diagram with scenarios, Class Diagram, State Diagram, Collaboration Diagram, Sequence Diagram, and Activity Diagram.
8. Defining Coding Standards and walkthrough.
9. Write the test cases for the identified module.
10. Demonstrate the use of different Testing Tools with comparison.
11. Define security and quality aspects of the identified module.

**Semester 5 - 4**

a. **Course Name:** Enterprise Programming using Java

b. **Course Code:** 303105309

c. **Prerequisite:** Basic knowledge of software applications

d. **Rationale:** This course helps students understand the principles of enterprise application development, database connectivity, server-side programming, and web application architecture. It enables students to design and develop scalable, secure, and platform-independent enterprise applications used in industries such as banking, e-commerce, and business management systems. The knowledge gained from this course prepares students for professional software development and enterprise system implementation.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Define the fundamentals of enterprise programming and use JDBC with Oracle and MySQL databases. |
| **CLOBJ 2** | Implement and configure servlets using configuration/context concepts, XML and annotations. |
| **CLOBJ 3** | Develop server-side applications using JSP, performing CRUD operations effectively. |
| **CLOBJ 4** | Explore and apply Hibernate ORM for database operations including annotations and queries. |
| **CLOBJ 5** | Design and implement applications using Spring framework and understand its architecture. |
| **CLOBJ 6** | Develop web applications using Spring Boot, including database interaction and REST services. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Analyze the structure and operations of JDBC for connecting and interacting with Oracle and MySQL databases. |
| **CLO 2** | Implement the concepts of Servlet Configuration and Context in practical scenarios. |

| | |
|---|---|
| **CLO 3** | Implement CRUD operations using JSP and Hibernate. |
| **CLO 4** | Design a web application using Spring Boot by developing controllers, services, and database integration to build scalable and functional web solutions. |

| Teaching Scheme | | | | Evaluation Scheme | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| L | T | P | C | Internal Evaluation | | | ESE | | Total |
| | | | | MSE | CE | P | Theory | P | |
| 2 | 0 | 2 | 3 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| Sr. No. | Content | Weightage | Teaching Hours |
|---|---|---|---|
| 1 | **Foundation of Enterprise Programming:** JDBC, JDBC architecture, JDBC with Oracle, MySQL, Maven: integration with Eclipse, POM.xml | 10% | 3 |
| 2 | **Servlets:** Basics of Web, Servlet Lifecycle, Servlets API, HTTP Servlets with XML and annotation, Servlets Configuration, Servlets Context, Servlets Collaboration, Session Tracking, CRUD operations | 15% | 4 |
| 3 | **JSP: Java Server Programming:** Scripting elements, Directive elements, CRUD operations. | 15% | 4 |
| 4 | **Hibernate (ORM):** Architecture, JPA, Generator class, Dialects, Mapping, Annotations, Transaction Management, HQL, HCQL, CRUD operations. | 20% | 6 |
| 5 | **Spring:** Architecture, Modules, Dependency Injection, Autowire, Application Context, annotation-based configuration, MVC CRUD operations | 20% | 7 |
| 6 | **Spring Boot:** Dependency Injection, Web App using Spring Boot, Spring Boot AOP, Spring Boot Database, Spring REST | 20% | 6 |

**i. Text Books and Reference Books:**


1. Java Enterprise in a Nutshell by Jim Farley, William Crawford, and David Flanagan (TextBook)
2. Java EE 8 Design Patterns and Best Practices by Rhuan Rocha

3. Java EE and HTML5 Enterprise Application Development by John Brock, Arun Gupta, and Geertjan Wielenga
4. Java 8 Programming Black Book

List of practical:

1. To study and implement basic Java programs and understand the setup of Java development environment for enterprise application development.
2. Write a Java program to demonstrate database connectivity using JDBC and perform basic database operations (Create, Insert, Update, Delete).
3. Develop a Java program using JDBC to retrieve and display records from a database table.
4. Create a simple Servlet program to demonstrate the handling of client requests and server responses.
5. Develop a Servlet application to process form data submitted by the user through an HTML page.
6. Write a JSP program to display dynamic content using JSP scripting elements.
7. Develop a JSP application that interacts with a database using JDBC to display records dynamically.
8. Create a web application using Servlets and JSP to implement user login and authentication system.
9. Develop a session management application using Servlets/JSP to maintain user session information.
10. Create a simple enterprise web application (such as Student Information System or Employee Management System) using JSP, Servlets, and JDBC.

**Semester 5-5**

a. **Course:** Professionalism & Corporate Ethics

b. **Course Code:** 303193304

c. **Prerequisite:** Knowledge of English language in practical life

d. **Rationale:** This course aims to provide students with a solid understanding of ethics in engineering, professionalism, and corporate ethics. It will help students develop communication skills, prepare for business school entrance exams, and improve their ability to make ethical decisions in a professional setting.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLO 1** | Discuss the scope and significance of engineering ethics and professionalism. |
| **CLO 2** | Apply ethical decision-making skills in resolving dilemmas in professional settings. |

| | |
|---|---|
| **CLO 3** | Develop communication and listening skills essential for professional environments. |
| **CLO 4** | Prepare for management entrance exams such as GMAT and CAT, and improve verbal communication. |
| **CLO 5** | Learn to effectively prepare business documents, such as brochures and minutes of meetings. |

**f. Course Outcome:**

| CO No. | Course Outcome Description |
|---|---|
| **CO 1** | Apply test-taking strategies effectively to prepare for the competitive examinations. |
| **CO 2** | Create brochures following standard layout, structure, and language |
| **CO 3** | Write clear and concise minutes for various types of meetings. |
| **CO 4** | Interpret audio and written materials using listening and reading comprehension strategies. |
| **CO 5** | Present viewpoints clearly and professionally in group discussions. |

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 0 | 1 | 0 | 1 | 0 | 100 | 0 | 0 | 0 | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|
| 1 | **Ethics in Engineering:** Scope of engineering ethics, accepting and sharing responsibility, resolving ethical dilemmas, case studies. | 20% | 5 |
| 2 | **Group Discussion:** Communication core, definition, types, process, guidelines, mock round-1. | 10% | 3 |
| 3 | **Introduction to B-School Tests:** Students will be able to solve verbal questions from the GMAT, CAT, and distinguish between national & international levels of Management exams. | 15% | 2 |
| 4 | **Listening Skills - Advanced Level:** Demonstrate ability to listen to more than two minutes of audio clips and solve questions based on it. | 10% | 1 |
| 5 | **Preparing Brochures:** Establishing the purpose of writing and determining the audience for whom the brochure is written. | 15% | 2 |
| 6 | **Agenda & Minutes of Meeting:** Explaining what an agenda and minutes of meeting are and their usefulness. | 10% | 1 |
| 7 | **Reading Comprehension - Intermediate Level:** Skim for main ideas, make use of contextual clues, and solve related questions. | 20% | 8 |

**Semester 5-6**

a. **Course:** Data Analytics and Data Visualization

b. **Course Code:** 303105314

c. **Prerequisite:** Database Management System, Linear Algebra, and Statistics.

d. **Rationale:** Data Analytics helps small and large organizations maximize the value of their data, unearth insights, build plans, and respond in real-time to customer demand.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLO 1** | Understand the core concepts of data visualization and data analytics, including the DIKW pyramid, infographics, and key differences between analysis and analytics. |
| **CLO 2** | Apply statistical techniques such as central tendency, variability measures, correlation, and histogram analysis to explore and interpret datasets. |

| | |
|---|---|
| **CLO 3** | Prepare and preprocess data by handling missing values, applying data cleaning techniques, and implementing dimensionality reduction using PCA and feature selection. |
| **CLO 4** | Implement regression models including simple, multiple, and logistic regression, and evaluate their effectiveness using metrics such as R-squared and SSE. |
| **CLO 5** | Use classification, clustering, and visualization tools like K-NN, decision trees, K-means clustering, and Power BI to extract insights and communicate findings interactively. |

**f. Course Outcome:**

| **CO No.** | **Course Outcome Description** |
|---|---|
| **CO 1** | Explain fundamentals of data visualization, DIKW pyramid, and applications of analytics. |
| **CO 2** | Apply descriptive and inferential statistics (central tendency, variance, correlation, distributions, CLT) for data analysis. |
| **CO 3** | Perform data preparation including cleaning, handling missing values, PCA, and feature selection. |
| **CO 4** | Implement regression techniques (simple, multiple, logistic) and evaluate with $R^2$, SSE, and related metrics. |
| **CO 5** | Use classification and clustering methods (KNN, decision trees, k-means) and tools (e.g., Power BI) for insights. |

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 3 | 0 | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|
| 1 | **Introduction:**<br>Introduction to data visualization & analytics, infographic representation of terminologies, DIKW (Data, Information, Knowledge, Wisdom) Pyramid, difference between analysis and analytics, applications of data visualization, applications of data analytics. | 15% | 7 |
| 2 | **Descriptive & Inferential Statistics:**<br>Population and sample, types of data (nominal, ordinal, discrete, continuous), measurement levels, representation of categorical variables, measures of central tendency (mean, median, mode), skewness, variance, standard deviation, coefficient of variation, covariance, correlation, histogram analysis, distribution & its types, central limit theorem. | 30% | 12 |
| 3 | **Data Preparation:**<br>Dealing with missing values, data cleaning using various methods, principal component analysis (PCA), feature selection methods. | 10% | 5 |
| 4 | **Regression:**<br>Application of regression for analytics, introduction to regression, simple and multiple linear regression, correlation vs. regression, SST (Sum of Squares Total), SSR (Sum of Squares Regression), SSE (Sum of Squares Error), RSquare, Adjusted R-Squared, logistic regression. | 25% | 11 |
| 5 | **Classification & Clustering:**<br>Use of classification & clustering for insights, K-NN, decision trees, K-means clustering, cluster analysis. Introduction to analytics tools like Power BI. | 20% | 10 |

**i. Reference Books:**


1. **The Art of Statistics: Learning from Data (Pelican Books)** (TextBook)
2. **Principles of Statistics** (TextBook) By M. G. Bulmer, Dover Publications Inc.

3. **Statistics 101: From Data Analysis and Predictive Modelling to Measuring Distribution and Determining Probability, Your Essential Guide to Statistics**
   By David Borman, Adams Media.
4. **Beautiful Visualization**
   By Noah Iliinsky, Julie Steele. Publisher(s): O'Reilly Media, Inc. ISBN: 9781449379865.

**j. List of Practicals:**

1. Use MS-Excel to create a pivot table & apply statistical measures to it.
2. Use the table created in the above practical to generate different charts.
3. Perform the Histogram Analysis of a given dataset using the Data Analysis Toolbox of Excel.
4. Use Python libraries to generate charts from data stored in Excel.
5. Perform Multiple Linear Regression on data.
6. Perform Logistic Regression on a dataset and interpret the regression table.
7. Use a dataset & apply K-NN to get insights from data.
8. Use a dataset & apply K-means clustering to get insights from data.
9. Study about tools like Orange, Tableau, Weka, etc., for data visualization.

Given a case study: Interactive Data Analytics with

## Semester 5-7

**a. Course:** AWS Fundamentals

**b. Course Code:** 303105301

**c. Prerequisite:** Basic understanding of computer concepts and basic programming

**d. Rationale:** This course provides a broad introduction to AWS cloud infrastructure, services, security and compliance, as well as billing, pricing, and support plans.

**e. Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Understand fundamental cloud computing concepts and the AWS platform. |
| **CLOBJ 2** | Configure and manage virtual networks (VPC) in AWS. |
| **CLOBJ 3** | Deploy and manage virtual machines (EC2 instances) in the AWS cloud. |

| | |
|---|---|
| **CLOBJ 4** | Utilize AWS storage services (S3, EBS) for various storage needs |
| **CLOBJ 5** | AwS Database services and Monitor AWS resources and applications |

**f. Course Outcome:**

| | |
|---|---|
| **CO 1** | Describe the fundamental cloud computing concepts and the AWS platform |
| **CO 2** | Analyze and manage virtual networks (VPC) in AWS. |
| **CO 3** | Analyze and manage virtual machines (EC2 instances) in the AWS cloud. |
| **CO 4** | Analyze AWS pricing strategies, support tiers, and budgeting tools to understand cost optimization in cloud environments. |
| **CO 5** | Apply knowledge of AWS SLAs, service lifecycle, and the Well-Architected Framework to assess and design resilient, efficient, and secure cloud architectures. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 2 | 0 | 0 | 2 | 20 | 20 | - | 60 | - | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|
| 1 | **Introduction to Cloud Computing and AWS**<br>What is Cloud Computing? . Key Benefits of Cloud Computing, Cloud Service Models, Cloud Deployment Models,<br>What is AWS?, The AWS Global Infrastructure. | 20% | 6 |
| 2 | **Networking Services**<br>Networking Basics,Virtual Private Cloud, subnets, route tables, security groups.<br>VPC Security,DNS:AWS Route 53,Cloud Front,Build Your VPC in AWS and Launch a Web Server. | 15% | 5 |
| 3 | **Compute Services**<br>IaaS, Virtual machines in the cloud,Amazon EC2, Elastic Load Balancing,AutoScaling, Launch an Amazon EC2 Serverless computing, AWS Lambda,Characterstics of AWS Lambda,Create and Configure a Lambda Function.Paas-AWS Elastic Beanstalk, Deploy a sample application to Elastic Beanstalk | 20% | 5 |
| 4 | **Storage Services**<br>Object Storage, Block Storage, and File Storage.,AWS Storage Services: Amazon S3,Various Storage classes of S3, Amazon EFS, and Amazon EBS,Create An amazon EBS volume. Hosting a Static Website using S3. | 15% | 5 |
| 5 | **AWS Architecture and Database Service**<br>AWS Well-Architected Framework Design Principles .Relational Database Service, MySQL, PostgreSQL, SQL Server, Amazon RDS (Relational Database Service), Amazon Aurora, Amazon DynamoDB, Amazon Redshift, Launch anAmazon RDS DB instance. | 15% | 5 |

| | | | |
|---|---|---|---|
| | **AWS Security , Monitoring, Scaling, and Billing** Identity and Access Management, users, groups, roles, policies,AWS IAM. Implement IAM Policies.Monitoring resources: AWS CloudWatch. Elastic Load Balancing ,Auto Scaling, , Cloud Economics and Billing, Scale and Load balance Your Architecture | 15% | 4 |

**i. Reference Books:**


1. **Cloud Computing: Concepts, Technology & Architecture:**By Thomas Erl, Ricardo Puttini, and Zaigham Mahmood"A Hands-On Guide to the Fundamentals of AWS Cloud" by Mark Wilkins

2. **AWS Certified Cloud Practitioner Study Guide: CLF-C01 Exam** By Ben Piper and David Clinton | McGraw Hill

3. **AWS Security: Identity and Access Management, Data Protection, and Application Security** By Mark Wilkins and Bryan Beausejour | Addison-Wesley Professional


**Semester 5-8**

**a. Course:** Cyber Security

**b. Course Code:** 30310504

**c. Prerequisite:** Fundamental of Programming, Computer Network

**d. Rationale:** Cyber security is the application of technologies, processes, and controls to protect systems, networks, programs, devices, and data from cyber-attacks. It aims to reduce the risk of cyber-attacks and protect against the unauthorized exploitation of systems, networks, and technologies.

**e. Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Explain the features and characteristics of the Linux Operating System and Windows Operating System. |
| **CLOBJ 2** | Apply network monitoring tools to identify attacks against network protocols and services. |
| **CLOBJ 3** | Apply various methods to prevent malicious access to computer networks, hosts, and data. |

| | |
|---|---|
| **CLOBJ 4** | Explain how to investigate endpoint vulnerabilities and attacks. |
| **CLOBJ 5** | Analyze network intrusion data to verify potential exploits. |
| **CLOBJ 6** | Apply incident response models to manage network security incidents. |

**f. Course Outcome:**

| | |
|---|---|
| **CO 1** | Explain information security concepts, threats, and risks. |
| **CO 2** | Use scanning tools (Nmap, Netcat, Wireshark) to detect vulnerabilities. |
| **CO 3** | Configure and analyze firewalls, VPNs, and IDS for defense. |
| **CO 4** | Classify cybercrimes and relate to laws (IT Act 2000) & forensics. |
| **CO 5** | Investigate attacks (malware, DoS, SQL injection, buffer overflow) and suggest mitigations. |
| **CO 6** | Explain information security concepts, threats, and risks. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 2 | 0 | 0 | 2 | 20 | 20 | - | 60 | - | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|
| 1 | **Information Security:** Introduction to information system, Types of information Systems, Development of Information Systems, Introduction to Information Security, Need for Information Security, Threats to Information Systems, Information Assurance, Cyber Security and Security Risk Analysis. | 15% | 6 |
| 2 | **Systems Vulnerability Scanning:** Overview of vulnerability scanning, Open Port/Service Identification, Banner/ Version Check, Traffic Probe, Vulnerability Probe, Vulnerability Examples. Networks Vulnerability Scanning- Netcat, Understanding Port and Services tools, Network Reconnaissance–Nmap. Network Sniffers and Injection tools–Wireshark. | 25% | 6 |
| 3 | **Network Defense tools:** Firewalls and Packet Filters: Firewall Basics, Packet Filter Vs Firewall, Firewall Protects a Network, Packet Characteristic to Filter, Stateless Vs Stateful Firewalls, Network Address Translation(NAT) and Port Forwarding, the basic of Virtual Private Networks, Linux Firewall, Windows Firewall, Snort: Introduction Detection System. | 20% | 6 |
| 4 | **Introduction to Cyber Crime and Law:** Cyber Crimes, Types of Cybercrime, Hacking, Attack vectors, Cyberspace and Criminal Behavior, Clarification of Terms, Traditional Problems Associated with Computer Crime, Introduction to Incident Response, Digital Forensics, Computer Language, Network Language, Realms of the Cyber world, A Brief History of the Internet, Recognizing and Defining Computer Crime, Contemporary Crimes, Computers as Targets, Contaminants and Destruction of Data, Indian ITACT 2000. | 20% | 4 |
| 5 | **Introduction to Cyber Crime Investigation:** Firewalls and Packet Filters, password Cracking, Keyloggers and Spyware, Virus And Worms, Trojan and backdoors, Steganography, DOS and DDOS attack, SQL injection, Buffer Overflow, Attack on wireless Networks. | 20% | 6 |

**i. Reference Books:**


1. " Cryptography and Network Security" by William Stallings — Pearson Education (Text Book)
2. " Anti-Hacker Tool Kit" by Mike Shema — Mc Graw Hill

3. ”Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives” by Nina Godbole and Sunit Belapure — WILEY
4. ”Cryptography and Network Security” by V.K. Jain — Khanna Publishing House
5. ”Information and Cyber Security” by Gupta Sarika — Khanna Publishing House
6. ”Cryptography and Network Security” by Atul Kahate — TMH

”Cryptography and Information Security” by V.K. Pachghare — PHI Learning

## Semester 5-9

a. **Course:** Internet of Things

b. **Course Code:** 303105305

c. **Prerequisite:** Basic Electronics and Circuits, Basic Programming Language.

d. **Rationale:** The explosive growth of the “Internet of Things” is changing our world. IoT components are allowing people to innovate new designs and products at home. This course will help students learn the importance of IoT in society, the current components of typical IoT devices, and trends for the future. It will also focus on the hardware and software components of embedded systems and the networking aspects of connecting devices to the internet.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Explain the technological trends leading to the development of IoT and its societal impacts. |
| **CLOBJ 2** | Define embedded systems and describe their interface with physical components. |
| **CLOBJ 3** | Identify and describe the key components of embedded systems used in IoT. |
| **CLOBJ 4** | Explain the interaction between hardware and software in IoT devices. |
| **CLOBJ 5** | Discuss and implement networking protocols for IoT, including UART and wireless sensor networks. |

f. **Course Outcome:**

| | |
|---|---|
| **CO 1** | Explain the fundamentals of embedded systems including microprocessors, microcontrollers, peripherals, and operating systems. |
| **CO 2** | Describe IoT concepts, devices, applications, benefits, risks, and security/privacy issues. |
| **CO 3** | Apply Arduino hardware and software platforms (IDE, shields, C operators, loops, and functions) to develop basic IoT applications. |
| **CO 4** | Demonstrate networking concepts, protocols, and Arduino-based serial communication for IoT connectivity. |
| **CO 5** | Integrate hardware and software components in IoT devices to interact with the physical world. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 2 | 0 | 0 | 2 | 20 | 20 | - | 60 | - | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage (%)** | **Teaching Hours** |
|---|---|---|---|
| 1 | **Introduction to IoT:**<br>Introduction to IoT, IoT Devices, IoT Devices vs. Computers, Societal Benefits of IoT, Risks, Privacy, and Security. | 20% | 6 |

| | | | |
|---|---|---|---|
| 2 | **Basics of Networking and Communication Protocols:** Need for Networking, Networking Components, Internet Structure, Protocols: UART and its Synchronization, Serial on Arduino, Reading from Serial, Introduction to Wireless Sensor Networking. | 20% | 6 |
| 3 | **IoT Hardware and Software:** Arduino Platform, Arduino IDE, Compiling Code, Arduino Shields, Arduino Basic Setup. Setting Up Your Environment, Variables, Basic C Operators, Conditionals, Loops, Functions, Global Variables. Python programming for IoT. Introduction to Raspberry Pi, Implementation of IoT with Raspberry Pi. | 30% | 8 |
| 4 | **Introduction to Embedded Systems:** Microprocessor, Microcontroller, GPU, I/O devices, clock, memory, other peripherals: ADC, DAC, Sensors and Actuators, Introduction to Operating Systems. | 20% | 6 |
| 5 | **Cloud Computing:** Fundamentals of Cloud computing, Cloud computing service models, Cloud computing management and security, IoT case studies. | 10% | 4 |

**i. Reference Books:**


1. "Internet of Things (A Hands-on Approach)" by Vijay Madisetti and Arshdeep Bahga, VPT
2. "Rethinking the Internet of Things: A Scalable Approach to Connecting Everything" by Francis daCosta, Apress Publications
3. "Embedded Systems: Architecture, Programming and Design" by Rajkamal, TMH
4. "Arduino Cookbook" by Michael Margolis, O'Reilly Publications
5. "Introduction to IoT" by S. Misra, A. Mukherji, and A. Roy, Cambridge University Press

**Semester 5-10**

a. **Course:** Fundamentals of Communication Engineering

b. **Course Code:** 303107346

c. **Prerequisite:** Fourier series, Fourier Transforms, Basic Electronics.

d. **Rationale:** This course explores the fundamentals of electronic communication systems. The course has two primary focuses: understanding electronic communication systems in analog form from a deterministic approach and gaining a broad understanding of satellite, optical, cellular, mobile, and wireless communications.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Define the basics and necessity of modulation in communication systems. |
| **CLOBJ 2** | Analyze various noise types and their effects on analog systems. |
| **CLOBJ 3** | Learn the principles of analog and digital modulation techniques. |
| **CLOBJ 4** | Explore fundamentals of local area networks and their hardware. |
| **CLOBJ 5** | Explain concepts of satellite, optical, wireless, and mobile communications. |

f. **Course Outcome:**

| | |
|---|---|
| **CO 1** | Explain the fundamental concepts and components of communication systems. |
| **CO 2** | Analyze and differentiate various types of modulation techniques used in communication systems. |
| **CO 3** | Apply communication modules in practical implementations and basic system designs. |
| **CO 4** | Describe the principles of wireless, cellular, and mobile communication systems. |
| **CO 5** | Explain the working principles of satellite and optical communication systems. |

g. **Teaching & Examination Scheme:**

| **Teaching Scheme** | **Evaluation Scheme** |
|---|---|

| L | T | P | C | Internal Evaluation | | | ESE | | Total |
|---|---|---|---|---|---|---|---|---|---|
| | | | | MSE | CE | P | Theory | P | |
| 2 | 0 | 0 | 2 | 20 | 20 | - | 60 | - | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|
| 1 | **Introduction:** Need for Modulation, Frequency translation, Electromagnetic spectrum, Gain, Attenuation and decibels. | 8% | 2 |
| 2 | **Noise:** Introduction, Thermal noise, Shot noise, Partition noise, Low frequency noise, Burst noise, a noise, High frequency noise, BJT and FET noises, Equivalent input noise generators, SNR, Tandem connection SNR, Noise factor and noise figure, Cascaded amplifiers, Lossy networks, Noise temperature, Measurement of noise, Narrow-band noise. | 20% | 3 |
| 3 | **Simple description on Modulation:** Analog Modulation (AM, FM), Pulse Modulation (PAM, PWM, PCM), Digital Modulation (ASK, FSK, PSK, QPSK). | 22% | 3 |
| 4 | **Networking and LAN:** Network fundamentals, LAN hardware, Ethernet LANs, Token Ring LAN. | 14% | 4 |
| 5 | **Satellite Communication & Optical Communication:** Satellite orbits, systems, subsystems, GPS, Optical principles, Fiber optic cables, WDM. | 22% | 6 |
| 6 | **Cellular, Mobile & Wireless Technologies:** Cellular systems: AMPS, GSM, CDMA, WCDMA. Wireless: WLAN, PAN, Bluetooth, WiFi, ZigBee, Mesh, WiMAX, MAN, Infrared, RFID, UWB, LTE, 5G. | 14% | 6 |

**i. Reference Books:**

1. Electronic Communications by Dennis Roddy & John Coolen — PHI
2. Electronic Communications by Kennedy — McGraw Hill Publication
3. Electronic Communications Systems by Wayne Tomasi — Pearson Education India
4. Electronic Communication Systems by Roy Blake — Cengage Learning
5. Communication Systems by Simon Haykins — Wiley India
6. Modern Digital and Analog Communication Systems by B. P. Lathi, Zhi Ding — Oxford University Press — 4th Edition
7. Wireless Communications Principles and Practice by T.S. Rappaport — PHI — 2nd edition
8. Introduction to Data Communications and Networking by Wayne Tomasi — Pearson Education
9. Theory and Problem of Electronic Communication by Lloyd Temes and Mitchel E.Schulz — McGraw Hill Publication

**Semester 6-1**

a. **Course:** Quant and Reasoning

b. **Course Code:** 303105311

c. **Prerequisite:** Good fundamentals in calculations and ability to think logically.

d. **Rationale:** The course focuses on building core aptitude and reasoning skills. These include analytical thinking, problem-solving, and logical decision-making abilities—vital for engineers in real-world scenarios such as software development, project analysis, and competitive assessments.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLO 1** | Define the core mathematical concepts related to number systems, averages, ratios, and profit/loss. |
| **CLO 2** | Apply logical reasoning techniques to problems involving directions, seating arrangements, syllogisms, clocks, and calendars. |
| **CLO 3** | Develop speed and accuracy in solving aptitude problems through practice and structured approaches. |
| **CLO 4** | Solve real-world problems using quantitative methods like permutations, combinations, and probability. |

| | |
|---|---|
| **CLO 5** | Prepare for competitive exams and placement tests by strengthening analytical and reasoning abilities. |

**f. Course Outcome:**

| | |
|---|---|
| **CO 1** | Apply logic and critical thinking skills to analyze information and draw logical conclusions. |
| **CO 2** | Solve complex problems by breaking them into manageable parts and creating effective solutions. |
| **CO 3** | Demonstrate the ability to approach problem-solving from different perspectives. |
| **CO 4** | Master foundational quantitative techniques for competitive and academic aptitude tests. |
| **CO 5** | Evaluate and solve diverse analytical puzzles and reasoning scenarios confidently. |

**g. Course Content:**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|
| 1 | **UNIT-1:** Number system, LCM & HCF simplifications and approximations | 9% | 4 |
| 2 | **UNIT-2:** Averages, progressions | 9% | 4 |
| 3 | **UNIT-3:** Ratio and proportion, Problems on Ages, Percentages | 12% | 5 |
| 4 | **UNIT-4:** Profit & loss, partnerships, S.I & C.I | 12% | 5 |
| 5 | **UNIT-5:** Time & work, pipes and cisterns, Time speed and distance, Problems on train crossings, Boats & streams | 18% | 8 |
| 6 | **UNIT-6:** Permutations & combinations, probability | 11% | 5 |
| 7 | **UNIT-7:** Directions, seating arrangements | 4% | 2 |
| 8 | **UNIT-8:** Clocks, calendars | 6% | 3 |
| 9 | **UNIT-9:** Cubes & Dice, syllogisms | 9% | 4 |

| | | | |
|---|---|---|---|
| 10 | **UNIT-10:**<br>Blood Relations | 5% | 2 |
| 11 | **UNIT-11:**<br>Series, Analogy, odd man out, coding and Decoding | 5% | 3 |

**h. Reference Books:**


1. "Quantitative Aptitude for CAT" by Arun Sharma (TextBook)
2. "Logical Reasoning for CAT" by Arun Sharma
3. "Quantitative Aptitude" by Abhijit Guha


## Semester 6-2

**a. Course:** Compiler Design

**b. Prerequisite:** Algorithms, Data Structures, Assembly Language Program, Theory of Computation, C/C++ Programming Skills

**c. Course Code:** 303105349

**d. Rationale:** Compiler Design is a fundamental subject of Computer Engineering. Compiler design principles provide an in-depth view of translation, optimization, and compilation of the entire source program. It also focuses on various designs of compiler and structuring of various phases of compiler. It is inevitable to grasp the knowledge of various types of grammar, lexical analysis, yacc, FSM (Finite State Machines), and correlative concepts of languages.

**e. Course Learning Objectives:**

| | |
|---|---|
| **CLO 1** | Define the structure and functions of a compiler including lexical analysis, syntax analysis, and semantic analysis. |
| **CLO 2** | Apply context-free grammars and parsing techniques (top-down and bottom-up) to analyze and construct syntax trees for programming languages. |
| **CLO 3** | Construct and evaluate syntax-directed definitions for translation and semantic analysis using S- and L-attributed grammars. |
| **CLO 4** | Generate intermediate code representations and apply basic code optimization strategies. |

| | |
|---|---|
| **CLO 5** | Explain runtime environment concepts including memory organization and activation records, and perform basic machine code generation. |

**f. Course Outcome:**

| CO No. | Course Outcome Description |
|---|---|
| **CO 1** | Describe compiler basics, lexical analysis, and syntax analysis using CFGs and parse trees. |
| **CO 2** | Explain top-down and LR parsing techniques, including predictive parsing, handle pruning, and YACC-based syntax analysis. |
| **CO 3** | Analyze and implement syntax-directed definitions using SDDs and attribute evaluation in parsers. |
| **CO 4** | Perform semantic analysis using symbol tables, scope handling, and S-/L-attributed SDDs for expressions, statements, and function declarations, including error recovery. |
| **CO 5** | Generate intermediate code for expressions, assignments, and control-flow constructs using various representations. |
| **CO 6** | Describe run-time environments, stack allocation, activation records, and implement basic machine code generation with optimizations. |

**g.Teaching & Examination Scheme:**

| Teaching Scheme | | | | Evaluation Scheme | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| L | T | P | C | Internal Evaluation | | | ESE | | Total |
| | | | | MSE | CE | P | Theory | P | |
| 3 | 0 | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**Course Content:**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|
| 1 | Overview of compilation:<br>The structure of a compiler and applications of compiler technology; Lexical analysis - The role of a lexical analyzer, specification of tokens, recognition of tokens, handwritten lexical analyzers, LEX, examples of LEX programs. | 10% | 8 |
| 2 | Introduction to syntax analysis<br>Role of a parser, use of context-free grammars (CFG) in the specification of the syntax of programming languages, techniques for writing grammars for programming languages (removal left recursion, etc.), non-context-free constructs in programming languages, parse trees and ambiguity, examples of programming language grammars. | 10% | 7 |
| 3 | Top-down parsing<br>FIRST & FOLLOW sets, LL(1) conditions, predictive parsing, recursive descent parsing, error recovery. LRparsing - Handle pruning, shift-reduce parsing, viable prefixes, valid items, LR(0) automaton, LR-parsing algorithm, SLR(1), LR(1), and LALR(1) parsing. YACC, error recovery with YACC and examples of YACC specifications. | 20% | 7 |
| 4 | Syntax-directed definitions (attribute grammars)<br>Synthesized and inherited attributes, examples of SDDs, evaluation orders for attributes of an SDD, dependency graphs. S-attributed and L-attributed SDDs and their implementation using LR-parsers and recursive-descent parsers respectively. | 15% | 6 |
| 5 | Semantic analysis<br>Symbol tables and their data structures. Representation of "scope". Semantic analysis of expressions, assignment, and control-flow statements, declarations of variables and functions, function calls, etc., using S- and L-attributed SDDs (treatment of arrays and structures included). Semantic error recovery. | 15% | 6 |

| | | | |
|---|---|---|---|
| 6 | Intermediate code generation<br>Different intermediate representations –quadruples, triples, trees, flow graphs, SSA forms, and their uses. Translation of expressions (including array references with subscripts) and assignment statements. Translation of control-flow statements – if-then-else, while-do, and switch. Short-circuit code and control-flow translation of Boolean expressions. Back patching. Examples to illustrate intermediate code generation for all constructs. | 15% | 6 |
| 7 | Run-time environments<br>Stack allocation of space and activation records. Access to non-local data on the stack in the case of procedures with and without nesting of procedures. | 10% | 3 |
| 8 | Introduction to machine code generation and optimization<br>Simple machine code generation, examples of machineindependent code optimizations. | 5% | 2 |

**Reference Books:**

**1.** Compilers: Principles, Techniques and Tools
By Aho, Lam, Sethi, and Ullman — Pearson — Second, Pub. Year 2014

**List of Practicals:**

**1.** Program to implement Lexical Analyzer.

**2.** Program to count digits, vowels and symbols in C.

**3.** Program to check validation of User Name and Password in C.

**4.** Program to implement Predictive Parsing LL (1) in C.

**5.** Program to implement Recursive Descent Parsing in C.

**6.** Program to implement Operator Precedence Parsing in C.

**7.** Program to implement LALR Parsing in C.

**8.** To Study about Lexical Analyzer Generator (LEX) and Flex (Fast Lexical Analyzer)

**9.** Implement following programs using Lex. • Create a Lexer to take input from text file and count no of characters, no. of lines & no. of words.

- Write a Lex program to count number of vowels and consonants in a given input string.

**10.** Implement following programs using Lex.

- Write a Lex program to print out all numbers from the given file.
- Write a Lex program to print out all HTML tags in file.
- Write a Lex program which adds line numbers to the given file and display the same onto the standard output.

**Semester 6-3**

a. **Course:** MEA(R)N Stack Web Development

b. **Course Code:** 303105385

c. **Prerequisite:** Database Management System, SQL, Basics of JavaScript and Web Development

d. **Rationale:**

(a) Understanding the fundamentals of JavaScript programming and web development

(b) Acquiring knowledge about how to store and retrieve data using MongoDB

(c) Acquiring knowledge about how to handle server-side logic and develop APIsusing Node.js, a server-side JavaScript runtime

(d) Learning how to create web apps with Express.js

(e) Acquiring knowledge about how to connect to APIs and create dynamic user interfaces using AngularJS, a potent front-end JavaScript framework

(f) Building a full-stack web application from scratch using the MEAN stack

(g) Understanding best practices for deploying, testing, and maintaining MEANstack applications

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLO 1** | Set up and configure the MEAN stack development environment by installing and managing MongoDB, Express.js, Angular, and Node.js. |
| **CLO 2** | Develop and manage databases using MongoDB by performing CRUD operations, creating indexes, and designing schemas for structured data storage. |
| **CLO 3** | Build server-side web applications using Node.js and Express.js, including API development, authentication, middleware handling, and security implementation. |
| **CLO 4** | Create dynamic front-end applications using Angular, with a focus on components, routing, data binding, forms, and HTTP communication. |

| | |
|---|---|
| **CLO 5** | Integrate, deploy, and optimize full-stack MEAN applications by connecting front-end and back-end systems, implementing real-time data features, and using cloud deployment practices. |

**f. Course Outcome:**

| **CO No.** | **Course Outcome Description** |
|---|---|
| **CO 1** | Explain fundamentals of web development and MEAN stack environment setup. |
| **CO 2** | Apply CRUD operations, indexing, and schema modeling using MongoDB. |
| **CO 3** | Implement server-side applications with Node.js and Express.js, including middleware, routing, and security. |
| **CO 4** | Develop dynamic front-end applications using Angular (components, services, data binding, routing, and forms). |
| **CO 5** | Integrate Angular front-end with Express APIs, authentication, and real-time data handling. |
| **CO 6** | Deploy MEAN applications with best practices (security, performance, version control) and demonstrate skills through a final project. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | 0 | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

**Course Content W - Weightage (%) , T - Teaching hours**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|
| 1 | **Introduction to Web Development and the MEAN Stack:** Overview of web development, Introduction to the MEAN stack, Setting up the development environment | 4% | 2 |
| 2 | **MongoDB:** Introduction to NoSQL databases, Installation and configuration of MongoDB, CRUD operations in MongoDB, Indexing and querying in MongoDB, Schema design and data modeling | 20% | 10 |
| 3 | **Node.JS & Express JS:** Introduction to Node.js and Express.js, Middleware and routing, Authentication and security with Passport.js, Error handling and logging | 20% | 10 |
| 4 | **Angular:** Introduction to Angular, Setting up an Angular application, Components, modules, and services, Data binding and templates, Forms and validation, Routing and navigation, HTTP and observables, Building a complete frontend for the MEAN stack application | 30% | 13 |
| 5 | **Integration:** Integrating the Angular frontend with the Express.js API, Authentication and user management integration, Handling real-time data with WebSockets, Error handling and testing | 10% | 3 |
| 6 | **Deployment and Best Practices:** Preparing the application for deployment, Hosting and server setup options, Security best practices, Performance optimization and testing, Version control and continuous integration | 6% | 3 |
| 7 | **Final Project:** Project | - | - |

**i. Reference Books:**

1. **MEAN Web Development**
   By Amos Q. Haviv — Packt Publishing (Textbook)
2. **Learning Node.js: A Hands-On Guide to Building Web Applicationsin JavaScript**
   By Marc Wandschneider — Addison-Wesley Professional
3. **AngularJS: Up and Running: Enhanced Productivity with Structured Web Apps**
   By Shyam Seshadri and Brad Green — O'Reilly Media
4. **MongoDB: The Definitive Guide: Powerful and Scalable Data Storage**
   By Shannon Bradshaw, Kristina Chodorow, and Eoin Brazil — O'Reilly Media

**j. List of Practicals:**

1. Introduction to MEAN stack, Setting up the development environment, Overview of MongoDB, Express.js, Angular, and Node.js.
2. Creating and configuring MongoDB, Creating and configuring Express.js, Building RESTful APIs with Express.js.
3. Introduction to Angular, Building basic UI components with Angular, Creating a Single-Page Application (SPA) with Angular.
4. Introduction to Node.js, Creating and configuring Node.js, Building server-side applications with Node.js.
5. Integrating all components to build a full-stack application, Testing and debugging the application, Deploying the application on a cloud platform.

**Semester 6-4**

a. **Course:** Employability Skills

b. **Course Code:** 303193353

c. **Prerequisite:** Basic knowledge of English communication and soft skills fundamentals.

d. **Rationale:** To enhance students' English proficiency, interpersonal communication, and job readiness through resume building, mock interviews, group discussions, and IELTS training.

e. **Learning Objectives:**

| | |
|---|---|
| **CLO 1** | Enhance English communication skills through IELTS-based listening, speaking, reading, and writing modules. |
| **CLO 2** | Develop a professional resume and cover letter tailored to job applications, higher education, or scholarships. |
| **CLO 3** | Strengthen interpersonal and critical thinking skills through mock group discussions on current and technical topics. |
| **CLO 4** | Prepare for personal and case interviews by understanding employer expectations and frequently asked questions. |
| **CLO 5** | Improve employability and global readiness through simulated workplace communication and assessment activities. |

f. **Course Outcomes:**

| | |
|---|---|
| **CO 1** | Demonstrate proficiency in IELTS-based communication skills applicable in global academic and professional settings. |
| **CO 2** | Create an effective resume and cover letter that align with specific job profiles or academic pursuits. |
| **CO 3** | Participate actively and confidently in structured group discussions, contributing valuable perspectives. |
| **CO 4** | Exhibit readiness for interviews by articulating thoughts clearly, addressing employer expectations, and handling case scenarios. |
| **CO 5** | Apply employability strategies and soft skills learned through mock assessments to real-world recruitment processes. |

**g. Teaching & Examination Scheme:**

| Teaching Scheme | | | | Evaluation Scheme | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| L | T | P | C | Internal Evaluation | | | ESE | | Total |
| | | | | MSE | CE | P | Theory | P | |
| - | 1 | - | 1 | - | 100 | - | - | - | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| Sr. No. | Topics | Weightage (%) | Teaching Hours(%) |
|---|---|---|---|
| 1 | IELTS Mock Test<br>To develop students English Learning and improve their employment prospects.<br>To create opportunity for students to study around the globe & give them Practice on :<br>Listening<br>Speaking<br>Reading<br>Writing | 25 | 5 |
| 2 | Resume Building<br>Cover letter & Resume Writing<br>Students will create a functional resume along with cover letter that they will be able to use when applying for a job, college or a scholarship. | 25 | 2 |
| 3 | Advanced Group Discussion: Mock Round<br>To provide students with an avenue to train themselves in various interpersonal skills.<br>To prepare students for the Group Discussion after the written test for employment or for admission to educational institutes.<br>To generate new ideas or new approaches for solving a problem.<br>To reach a solution on an issue of concern. | 25 | 4 |

| | | | |
|---|---|---|---|
| 4 | Personal Interview: Mock Round<br>Preparing For The Interview<br>Review Question<br>Employer's Expectation<br>Case Interview | 25 | 4 |

**i. Reference Books:**


1. **Business Correspondence and Report Writing** by R. Sharma and K. Mohan
2. **Communication Skills and Soft Skills** by Suresh Kumar, Pearson Publication, 2010

**Semester 6-5**

a. **Course:** Artificial Intelligence

b. **Prerequisite:** Basic understanding of algorithms and probability theory.

c. **Course Code:** 303105307

d. **Rationale:** To provide foundational knowledge and hands-on experience in Artificial Intelligence, covering search techniques, reasoning, learning, expert systems, and neural networks.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLO 1** | Describe the foundational concepts, problem-solving approaches, and major application areas of Artificial Intelligence. |
| **CLO 2** | Apply various AI search strategies (e.g., BFS, DFS, A*, AO*) and gameplaying algorithms (like Minimax and Alpha-Beta pruning) to solve classic AI problems efficiently. |
| **CLO 3** | Demonstrate knowledge representation using predicate calculus, semantic networks, production rules, and apply reasoning methods for logical inference and decision-making. |
| **CLO 4** | Design and implement learning models including rote learning, inductive learning, and neural networks (supervised, unsupervised) to solve datadriven problems. |
| **CLO 5** | Analyze and apply uncertain reasoning techniques such as Bayesian networks, fuzzy logic, and expert systems to real-world scenarios in intelligent decision-making. |

f. **Course Outcome:**

| **CO No.** | **Course Outcome Description** |
|---|---|
| **CO 1** | Describe AI fundamentals, major areas, applications, and problem representation using state space search. |
| **CO 2** | Apply various search techniques, including BFS, DFS, heuristic, and constraint-based methods, to solve AI problems. |
| **CO 3** | Utilize game-playing strategies, including Minimax and Alpha-Beta pruning, to optimize decision-making in AI applications. |

| | |
|---|---|
| **CO 4** | Demonstrate AI knowledge representation using rules, calculus, semantic nets, frames, and reasoning methods. |
| **CO 5** | Illustrate AI learning methods and handle uncertainty using probabilistic and fuzzy reasoning. |
| **CO 6** | Explain expert systems and connectionist models, including neural networks, their learning rules, and applications. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | 0 | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage (%)** | **Teaching Hours** |
|---|---|---|---|
| 1 | Introduction<br>What is artificial Intelligence? Major areas of Artificial Intelligence, Introduction to AI Problems and applications, Defining problems as a state space search, Production systems. | 10% | 4 |
| 2 | Search techniques<br>Breadth first search, Depth first search, Hill climbing, Best first search, A* algorithm, AO* Algorithm, Iterative Deepening Search, IDA*, Recursive Best First Search, Constraint Satisfaction and Heuristic Repair, Applications | 20% | 4 |
| 3 | Game Playing:<br>Introduction to Game playing, The Minimax Search Procedure, AlphaBeta Procedure, The Search Efficiency of Alpha-Beta Procedure, Recent applications | 10% | 4 |

| | | | |
|---|---|---|---|
| 4 | Knowledge Representation<br>Production rules, Predicate Calculus- Rules of Inference; Semantics and Deduction; Unification; Soundness and completeness of rules; Resolution; Resolution refutation, Semantic Nets, Frames, symbolic reasoning, statistical reasoning. | 10% | 6 |
| 5 | Learning<br>Definition, Rote learning, learning by taking advice, learning in problem solving, learning from examples, induction | 10% | 6 |
| 6 | Uncertain Reasoning<br>Joint probability, Marginal probability, Probabilistic reasoning and Bayes Nets, forward reasoning versus backward reasoning, Certainty Factors, Fuzzy set theory, Fuzzy relation, fuzzification, Fuzzy value assignment methods, Inference and Composition methods- Min-Max composition, max product composition, Defuzzification methods, Applications and recent developments | 20% | 6 |
| 7 | Expert Systems (ES)<br>Advantages and characteristics of Expert System, Knowledge engineering, Steps in Developing an Expert System, Mycin, ES Applications and recent developments. | 10% | 6 |
| 8 | Connectionist Models<br>Introduction to Neural Network, Activation functions, Supervised and Unsupervised Learning, Neuro Processing and Neural Network Learning, Learning, Learning rules, Single layer Perceptrons and Classification, Introduction to Multilayer Neural Networks, Neural Network Applications and recent developments | 10% | 6 |

**i. Reference Books:**


1. "Artificial Intelligence" By Elaine Rich and Kevin Knight — TMH
2. "Artificial Intelligence: A New Synthesis" By N. J. Nilsson — Harcourt Publishers
3. "Fuzzy Logic and Engineering Application" By Tomthy Ross — Wiley Publication.
4. "Expert Systems Principles and Programming" By Giarratano & Riley son — Vikas Publishing House — 3rd Edition
5. "Elements of Artificial Neural Network" By Kishan Mehrotra
6. "Genetic Algorithms in search, Optimization and Machine" By Goldberg D. E Addison — Wesley New York
7. "Neural Networks" By J. M. Jurada

**j. List of Practicals:**

1. Write a program to implement Tic Tac Toe game.
2. Write a program to implement 8 Puzzle problems.
3. Write a program to implement Water Jug Problem.
4. Write a program to implement Travelling Salesman Problem.
5. Write a program to implement N Queens Problem.
6. Write a program to implement Tower of Hanoi Problem.
7. Write prolog programs for following problems.
8. Demonstrate Knowledge Base and Query System in prolog.
9. Convert Prolog predicates into Semantic Net.
10. Demonstrate supervised learning using artificial neural network.

**Semester 6-6**

a. **Course:** Machine Learning

b. **Course Code:** 303105353

c. **Prerequisite:** Basic knowledge of programming in Python, Fundamental concepts of Mathematics, including Linear Algebra, Probability, and Statistics, Understanding of Data Structures and Algorithms for efficient data handling and processing.

d. **Rationale:** The course aims to provide a comprehensive understanding of Machine Learning (ML) concepts, including supervised, unsupervised, and reinforcement learning, develop proficiency in decision tree learning, neural networks, and Bayesian learning, apply genetic algorithms, fuzzy logic, and optimization techniques to real-world problems, enable students to analyze and compare ML models based on accuracy, efficiency, and application, and implement ML models using Python/Java for text classification, face recognition, healthcare, and fraud detection.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLO 1** | Explain key paradigms in machine learning including supervised, unsupervised, semi-supervised, and reinforcement learning. Develop the ability to differentiate among these approaches and identify suitable real-world applications such as spam detection, customer segmentation, and game AI. |
| **CLO 2** | Design and implement fundamental classification algorithms such as Decision Trees, K-Nearest Neighbors (KNN), and Na¨ıve Bayes. Evaluate their performance using appropriate metrics and apply them to structured data problems in fields like finance, healthcare, and e-commerce. |
| **CLO 3** | Develop and train artificial neural networks (ANNs) including multi-layer perceptrons. Apply neural network-based learning and genetic algorithms to solve complex tasks such as image-based face recognition, robotic control, and medical diagnosis. |
| **CLO 4** | Analyze probabilistic models in Bayesian learning. Implement classifiers like Na¨ıve Bayes and Bayesian Belief Networks and apply ExpectationMaximization (EM) algorithm for handling incomplete data. Apply these techniques to domains such as natural language processing and diagnostics. |
| **CLO 5** | Investigate soft computing approaches including fuzzy logic systems and optimization algorithms like Genetic Algorithms, Ant Colony Optimization (ACO), and Particle Swarm Optimization (PSO). Apply them in solving real-world optimization problems such as intrusion detection, scheduling, fraud detection, and healthcare decision systems. |

**f. Course Outcome:**

| CO No. | Course Outcome Description |
|---|---|
| **CO 1** | Explain fundamental concepts of machine learning and concept learning techniques. |
| **CO 2** | Apply supervised and unsupervised learning techniques, including decision trees, k-NN, radial basis functions, and case-based reasoning. |
| **CO 3** | Utilize neural networks and backpropagation for pattern recognition. |
| **CO 4** | Apply Bayesian learning techniques, including Bayes theorem, Naïve Bayes, and Bayesian networks, for classification problems. |
| **CO 5** | Implement fuzzy logic concepts, including fuzzy rule-based systems, decision making, classification, and pattern recognition. |
| **CO 6** | Employ optimization techniques, including genetic algorithms, ant colony optimization, and particle swarm optimization, for real-world applications. |

**g. Teaching & Examination Scheme:**

| Teaching Scheme | | | | Evaluation Scheme | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| L | T | P | C | Internal Evaluation | | | ESE | | Total |
| | | | | MSE | CE | P | Theory | P | |
| 3 | 0 | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|

| | | | |
|---|---|---|---|
| 1 | **Introduction:**<br>Learning Problems, designing a learning system, Issues with machine learning. Concept Learning, Version Spaces and Candidate Eliminations, Inductive bias. | 10% | 5 |
| 2 | **Supervised and Unsupervised learning:**<br>Decision Tree Representation, Appropriate problems for Decision tree learning, Algorithm, Hypothesis space search in Decision tree learning, inductive bias in Decision tree learning, Issues in Decision tree learning ,K- Nearest Neighbour Learning, Locally Weighted Regression, Radial Bases, Functions, Case Based Reasoning. | 24% | 12 |
| 3 | **Artificial Neural networks and genetic algorithms:**<br>Neural Network Representation, Appropriate problems for Neural Network Learning, Perceptrons, Multilayer Networks and Back Propagation, Algorithms, Remarks on Back Propagation Algorithms, Case Study: face Recognition. | 18% | 7 |
| 4 | **Bayesian Learning:**<br>Bayes Theorem, Bayes Theorem and Concept Learning, Maximum Likelihood and Least squared Error Hypothesis, Bayes Optimal Classifier, Gibbs Algorithm, Na¨ıve Bayes Classifier, Bayesian Belief Network, EM Algorithm Case Study: Learning to classify text. | 18% | 7 |
| 5 | **Fuzzy Logic:**<br>Classical Logic and Fuzzy logic, Fuzzy Rule based systems, Fuzzy Decision making, Fuzzy Classification, Fuzzy Pattern Recognition, Applications. | 20% | 9 |
| 6 | **Optimization Techniques:**<br>Derivative based Optimization – Descent Methods – Genetic Algorithms – Ant Colony Optimization – Particle Swarm Optimization, Case Study - fraud detection, health care using Soft computing techniques. | 10% | 5 |

**i. Reference Books:**


1. **Real-World Machine Learning** (Text Book)
   By Henrik Brink, Joseph Richards, Mark Fetherolf — Dream Tech
2. **Pattern Recognition and Machine Learning**By Bishop, Christopher — Springer
3. **Elements of Statistical Learning**
   By Hastie, Tibshirani, and Friedman — Soft Computing for Problem Solving, AISC, Springer
4. **Data Mining: Tools and Techniques**By Jiawei Han and Michelline Kamber

5. **Data Mining: A practical Machine Learning Tools and techniques**By IH Witten, Eibe Frank, Mark A Hall — Elsevier

**j. List of Practicals:**

1. Dealing with Data using Numpy, Pandas, Statistics library
2. Data Analysis & Visualization on Diwali Sales Dataset.
3. Implement linear regression and logistic regression.
4. Implement the na¨ive Bayesian classifier for a sample training dataset stored as a .CSVfile. Compute the accuracy of the classifier, considering a few test data sets.
5. Assuming a set of documents that need to be classified, use the naive Bayesian Classifier model to perform this task.
6. Decision tree-based ID3 algorithm.
7. Write a program to implement the K-Nearest Neighbor algorithm to classify the iris data set
8. Apply EM algorithm to cluster a set of data stored in a .CSVfile. Use the same data set for clustering using k-Means algorithm.
9. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set.
10. Compare the various supervised learning algorithm by using appropriate dataset. (Linear Regression, Support Vector Machine, Decision Tree)
11. Compare the various Unsupervised learning algorithm by using the appropriate datasets. (K Means Clustering, K Mode)
12. Build an Artificial Neural Network by implementing the Back propagation algorithm and test the same using appropriate data sets.

**Semester 6-7**

a. **Course Name:** Cloud Computing

b. **Course Code:** 303105363

c. **Prerequisite:** Fundamentals of Distributed Computing

d. **Rationale:** This course aims students to understand the hardware, software concepts and architecture of cloud computing. Students realize the importance of Cloud Virtualization, Abstractions and Enabling Technologies.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Compare the strengths and limitations of cloud computing. |
| **CLOBJ 2** | Identify the architecture, infrastructure and delivery models of cloud computing. |
| **CLOBJ 3** | Apply suitable virtualization concepts. |
| **CLOBJ 4** | Choose the appropriate cloud player, programming models and approach. |
| **CLOBJ 5** | Address the core issues of cloud computing such as security, privacy and interoperability. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Describe the fundamentals of cloud computing, its layers, types, infrastructure management, challenges, and applications. |
| **CLO 2** | Explain virtualization of computing, storage, and resources, and illustrate cloud service models including IaaS, PaaS, and SaaS. |

| | |
|---|---|
| **CLO 3** | Explain the concept of Infrastructure as a Service (IaaS), including hypervisors, resource virtualization, examples, and implementation approaches. |
| **CLO 4** | Apply Platform as a Service (PaaS) and Software as a Service (SaaS) concepts to analyze examples and demonstrate practical implementation in cloud environments. |
| **CLO 5** | Explain cloud service management, SLAs, billing, scaling, and performance evaluation in real-world scenarios. |
| **CLO 6** | Analyze cloud security challenges, data protection mechanisms, legal considerations, and assess real-world cloud platforms such as Eucalyptus, VMware, IBM Bluemix, Google Cloud, and AWS. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | 0 | 2 | 4 | 20 | 20 | - | 60 | - | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage (%)** | **Teaching Hours** |
|---|---|---|---|
| 1 | Introduction: Cloud Computing, Layers and Types of Clouds, Cloud Infrastructure Management, Challenges and Applications. | 15% | 5 |
| 2 | Virtualization: Virtualization of Computing, Storage and Resources. Cloud Services: Introduction to Cloud Services IaaS, PaaS and SaaS | 15% | 5 |

| | | | |
|---|---|---|---|
| 3 | Infrastructure as a Service (IaaS): Introduction, Hypervisors, Resource virtualization, Examples, How to implement IaaS | 10% | 5 |
| 4 | Platform as a Service (PaaS): Introduction, Cloud Platform and Management, Examples, How to implement PaaS | 10% | 5 |
| 5 | Software as a Service (SaaS): Introduction, Web services, Web 2.0, Web OS, Examples, How to implement SaaS | 10% | 5 |
| 6 | Service Management in Cloud Computing: Service Orchestration, SLAs, Billing & Accounting, Scaling hardware, Economics of Scaling, Managing Data, Performance, Project Experiences | 15% | 7 |
| 7 | Security: Cloud Storage from LANs to WANs, Data Security Technologies, Security Concerns, Legal Issues, Securing Private and Public Clouds | 15% | 7 |
| 8 | Case Study: Eucalyptus, VMware, IBM Bluemix, Google Cloud, Amazon Web Services | 10% | 6 |

**i. Reference Books:**


1. **Cloud Computing Bible** by Barrie Sosinsky, Wiley India Pvt Ltd (TextBook)
2. **Cloud Computing Principles and Paradigms** by Rajkumar Buyya, James Broberg, Andrzej Goscinski, Wiley
3. **Cloud Computing: Principles, Systems and Applications** by Nikos Antonopoulos, Lee Gillam, Springer, 2012
4. **Cloud Security and Privacy: An Enterprise Perspective on Risks andCompliance** by Tim Mather, Subra Kumaraswamy, Shahed Latif, O'Reilly Media, 2009
5. **Cloud Security: A Comprehensive Guide to Secure Cloud Computing** by Ronald L. Krutz, Russell Dean Vines, Wiley-India, 2010


**List of Practicals:**

1. Understanding single core and multi-core architecture.
2. Understanding computer network fundamentals and designing LANs.
3. Implementation of Infrastructure as a Service (IaaS) using Hypervisors.
4. Implementation of private cloud platform using OpenStack cloud.
5. Working with IaaS of public cloud platforms.
6. Implementation of Platform as a Service (PaaS) in private cloud environment.

7. Implementation of Platform as a Service (PaaS) in public cloud environment.
8. Implementation of Software as a Service (SaaS) in private cloud environment.
9. Implementing Software as a Service (SaaS) in public cloud environment.
10. Implementation of Storage as a Service (SaaS).

## Semester 6-8

a. **Course:** Mobile App Development

b. **Course Code:** 303105379

c. **Prerequisite:** Basic knowledge of java language

d. **Rationale:** The mobile application development syllabus covers the essential concepts and tools for building apps across platforms, including UI/UX design, app architecture, networking, databases, and deployment. It explores both native development (Android) and cross-platform frameworks, emphasizing practical skills for creating functional, user-friendly mobile applications.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLO 1** | Acquire an insight into concepts of mobile application development terminologies, environment and architecture |
| **CLO 2** | Design mobile application using various UI components and layouts. |
| **CLO 3** | Develop robust mobile applications with database interaction and webservice integration |
| **CLO 4** | Deploy application on mobile device |

f. **Course Outcome:**

| **CO No.** | **Course Outcome Description** |
|---|---|
| **CO 1** | Acquire an insight into concepts of mobile application development terminologies, environment and architecture |
| **CO 2** | Design mobile application using various UI components and layouts. |

| | |
|---|---|
| **CO 3** | Develop robust mobile applications with database interaction and webservice integration |
| **CO 4** | Deploy application on mobile device |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | 0 | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

| **Sr. No.** | **Topics** | **Weightage (%)** | **Teaching Hours** |
|---|---|---|---|
| 1 | **Android Operating System and Development Environment :** Introduction, Android Architecture, Versions, Features, OHA, Dalvik VM, Android SDK, Android Development Tools, Android Virtual Devices, Development Environment, Directory Structure of Android Application, Android Manifest file | 10% | 3 |
| 2 | **Android Components and Resource handling :** Components: Context, Activity, Intent, Service, Broadcast Receiver, Resources:String, Color, Drawable, Styles, Theme, Localization:Prepare Application for Localization | 20% | 7 |
| 3 | **Android User Interface Elements and Layouts:** Introduction of Material Design, UI and UX Layouts: Linear Layout, Absolute Layout, Frame Layout, Relative Layout, Constraint Layout, Dynamic Implementation of Layout. UI widgets | 20% | 8 |

| | | | |
|---|---|---|---|
| | with properties, events and methods, Dialog boxes, Menus: Option and Context | | |
| 4 | **Working with Views and Fragment:** GridView, WebView, ScrollView, ListView, RecyclerView, CardView Fragment: Introduction, life Cycle, Implementation | 10% | 05 |
| 5 | **Data Storage Techniques :** Shared Preferences, Files and Directories, SQLite Database Connectivity andOperations, Content Providers: Basics, Content URI, Content Resolver, Built-in content providers. | 20% | 09 |
| 6 | **Web Application Integration Techniques:** Introduction of AsyncTask, Communication with Web API, Introduction to JSON data, JSON Parsing,Implementation of Third-PartyLibraryto Fetch Network Data, Notifications, Telephony API, Google API | 10% | 08 |
| 7 | **Polish and Publish Application:** Different Ways to Monetize, Versioning, Signing, Packaging and Beta Test of Mobile Application, Distributing Application on Mobile Market Place | 10% | 05 |

**i. Reference Books:**


1. **Android Wireless Application Development By Lauren Darcey and Shane Conder | Pearson Education,2011 | second edition (TextBook)**
2. **Head First Android Development: A Brain Friendly Guide, O`Reilly, David Griffiths and Dawn Griffith**
3. **Professional Android 4 Application Development, John Wiley & Sons Author(s): Reto Meier**


**j. List of Practicals:**

1. Create a "Hello World" application: Display "Hello World" at the center of the screen, both on the Android emulator and an actual Android device.

2. Build an app to showcase Android lifecycle phases: Develop an app that demonstrates various Android lifecycle stages (onCreate, onStart, onResume, etc.).

3. Create an app with two activities: The first activity should contain an EditText and a "Send" button. When the button is clicked, use an explicit intent to send the text from EditText to a second activity and display it in a TextView.

4. Create an app with explicit intent: The first activity should have an EditText and a "Send" button. On button click, use an implicit intent with the "SEND" action, allowing the user to select an app from an app chooser to handle the intent and display the text.

5. Build a basic calculator app: Create an app that performs basic arithmetic operations (addition, subtraction, multiplication, and division) on numbers.

6. Create a Spinner-based app: Develop an app with a spinner populated from the res/values/strings.xml resource. When the spinner value changes, the corresponding image from the res/drawable directory should be displayed.

7. Create a discount calculator app: Use a RadioGroup with three radio buttons for 10%, 15%, and 20% discounts on a shopping bill. The user can enter the bill amount in an EditText, and the selected discount will be calculated and displayed in a TextView.

8. Create an app with a course selection RadioButton group: Display a list of college courses with a RadioButtongroup. When a course is selected, the corresponding TIC (Total Instructional Credit) should be shown in a TextView.

9. Create a shopping list app using checkboxes: Build an app with checkboxes for shopping list items. As items are checked off, the selected items should be displayed in a TextView.
10. Create a login and registration app: Develop a login application that verifies the username and password. Include a registration page for new users. Upon successful login, show a "Welcome User" pop-up message

11. Create a login app with navigation to another activity: The login screen should verify the username and password. After successful login, navigate to a new activity that displays a "Welcome User" message in a TextView and a "Logout" button. On clicking "Logout," show a confirmation dialog with "OK" and "Cancel" buttons. "OK" should return to the login screen, while "Cancel" should keep the user on the current activity.

12. Create an app with a menu: Implement a menu with five options. The selected option should be displayed in a TextView.

13. Build an app using LinearLayout: Create a simple app that uses LinearLayout. It should take the contents of a predefined TextView, convert it to uppercase on button click, and display it in an EditText. Additionally, create an app that responds to key events in the EditText without needing a button press.

14. Create an app with TableLayout and custom styles: Use a TableLayout with a TextView, EditText, and buttons. Also, create a custom styles.xml in the res/values directory to style the TextView.

15. Create an app with SQLite database operations: Build an app that allows the user to perform CRUD operations (Create, Read, Update, Delete) with an SQLite database.

16. Create an app with three vertically aligned buttons: Develop an app with three buttons arranged vertically. When any button is selected, the screen color should change accordingly

**Semester 6-9**

a. **Course Name:** .NET Programming

b. **Course Code:** 303105351

c. **Prerequisite:** Concepts of Object-oriented programming approach — 203105102 - Programming for Problem Solving

d. **Rationale:** This is an introductory programming course using the C# language. It does not assume any prior programming experience. This course will prepare students for intermediate C# and ASP.NET courses. This is an optional course in the Local Area Network Administration and Microcomputer Applications Support AAS degrees, and in the Local Area Network Administration and Database Certificates.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Define the fundamentals of C# programming including data types, control structures, and object-oriented programming concepts. |
| **CLOBJ 2** | Explore the .NET framework and its components for application development. |
| **CLOBJ 3** | Practice the development of simple console-based C# applications. |
| **CLOBJ 4** | Develop and debug C# programs using the .NET compiler tools. |
| **CLOBJ 5** | Demonstrate confidence in building desktop and web-based applications using C# and .NET technologies. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Apply OOP concepts to develop C# console applications. |

| | |
|---|---|
| **CLO 2** | Develop .NET projects using assemblies, namespaces, exceptions, I/O, and collections. |
| **CLO 3** | Implement data connectivity using ADO.NET with datasets and data binding. |
| **CLO 4** | Design user interfaces with Windows Forms and apply visual inheritance. |
| **CLO 5** | Build dynamic web apps using ASP.NET master pages, themes, and GridView. |
| **CLO 6** | Manage state in web apps and integrate XML web services. |

**Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | 0 | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination **Course Content:**

| **Sr. No.** | **Topics** | **Weightage (%)** | **Teaching Hours** |
|---|---|---|---|
| 1 | Introduction of C#:<br>C# The Basics and Console Applications in C#: Name Spaces - Constructor and Destructors, Function Overloading & Inheritance, Operator Overloading, Modifiers - Property and Indexers, Attributes & Reflection API, When to use Console Applications. Generating Console Output, Processing Console Input. | 20% | 8 |

| | | | |
|---|---|---|---|
| 2 | C#.NET:<br>C#.NET Language Features and Creating .NET Projects, Namespaces Classes and Inheritance, Exploring the Base Class Library, Debugging and Error Handling, Data Types, Exploring Assemblies and Namespaces, String Manipulation, Files and I/O, Collections. | 10% | 4 |
| 3 | ADO.NET:<br>Benefits of ADO.NET, ADO.NET compared to classic ADO, Datasets, Managed Providers, Data Binding: Introducing Data Source Controls, Reading and Writing Data Using the SqlDataSource Control. | 15% | 6 |
| 4 | Windows Forms and Controls in details:<br>The Windows Forms Model, Creating Windows Forms, Windows Forms Properties and Events, Windows Form Controls, Menus, Dialogs, ToolTips. | 10% | 4 |
| 5 | Visual Inheritance in C#.NET:<br>Apply Inheritance techniques to Forms, Creating Base Forms, Programming Derived Forms. | 10% | 4 |
| 6 | Themes and Master Pages:<br>Creating a Consistent Web Site, ASP.NET 2.0 Themes Master Pages, Displaying Data with the GridView Control, Introducing the GridView Control, Filter Data in the GridView Control, Allow Users to Select from a DropDownList in the Grid, Add a Hyperlink to the Grid, Deleting a Row and Handling Errors. | 10% | 5 |
| 7 | Managing State:<br>Preserving State in Web Applications and Page-Level State, Using Cookies to Preserve State, ASP.NET Session State, Storing Objects in Session State, Configuring Session State, Setting Up an Out-of-Process State Server, Storing Session State in SQL Server, Using Cookieless Session IDs, Application State Using the DataList and Repeater Controls, Overview of List-Bound Controls, Creating a Repeater Control and DataList Control. | 20% | 7 |
| 8 | Creating and Consuming Web Services:<br>The Motivation for XML Web Services, Creating an XML Web Service with Visual Studio, Designing XML Web Services, Creating Web Service Consumers, Discovering Web Services Using UDDI. | 5% | 7 |

**Reference Books:**

**1.** Christian Nagel, *Professional C# .Net*, Wrox Publication. (TextBook) **2.** Matthew Macdonald and Robert Standefer, *ASP.NET Complete Reference*, TMH.

**3.** Vijay Mukhi, *C# The Basics*, BPB Publications.

**List of Practicals:**

1. Write a program to Enable-Disable Textbox and change width of TextBox.
2. Write a program to increase and decrease font size programmatically.
3. Write C# code to display the asterisk pattern as shown below: ***** *****
4. Write C# code to prompt a user to input his/her name and country name and then the output will be shown as an example below: Hello Ram from country India!
5. Write C# code to do the following • Convert binary to decimal • Convert decimal to hexadecimal • Convert decimal to binary • Convert decimal to octal
6. Write C# code to convert infix notation to postfix notation.
7. Write a C# code to convert digits to words.
8. Write a C# code to Convert following currency conversion. Rupees to dollar, frank, euro.
9. Write a C# code to Perform Celsius to Fahrenheit Conversion and Fahrenheit to Celsius conversion.
10. Write ASP.Net program to Store Objects in Session State and Storing Session State in SQLServer.

**Semester 6-10**

a. **Course:** DevOps

b. **Course Code:** 303105387

c. **Prerequisite:** Basic knowledge of software development and operations.

d. **Course Objective:** This course provides a broad introduction to software development and operations in DevOps. The various process models required to develop software applications are also described. The improvement and the collaboration between developers and operators are also described. The students will learn how DevOps helps the software development life cycle and how to manage the infrastructure using automation tools and code.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLO 1** | Explain DevOps core concepts, principles, workflow, Agile comparison, and the roles of a DevOps engineer. |
| **CLO 2** | Use infrastructure automation tools like AWS, Chef, Puppet, Jenkins, Splunk, AppDynamics, and Nagios for configuration and deployment. |
| **CLO 3** | Work with Docker and Jenkins to containerize applications and automate build-deploy processes. |
| **CLO 4** | Automate testing and CI/CD pipelines using Jenkins with Selenium and TestNG for continuous integration and delivery. |
| **CLO 5** | Perform Git operations like branching, merging, stashing, and collaborate on projects using GitHub. |

f. **Course Outcome:**

| **CO No.** | **Course Outcome Description** |
|---|---|
| **CO 1** | Explain the fundamentals of DevOps, its principles, workflows, roles, and significance compared to Agile and traditional IT practices. |
| **CO 2** | Demonstrate the use of DevOps automation tools (AWS, Jenkins, Puppet, Chef, Splunk, Nagios, App Dynamics) for infrastructure, configuration, deployment, and monitoring. |
| **CO 3** | Apply containerization concepts using Docker, including architecture, installation, and provisioning, and manage container clusters using Docker Swarm. |

| | |
|---|---|
| **CO 4** | Implement Continuous Integration/Continuous Deployment (CI/CD) pipelines using Jenkins and perform automated testing with TestNG, Selenium, and batch scripts. |
| **CO 5** | Utilize Git for version control, branching strategies, and collaborative project management on GitHub. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | 0 | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

**Course Content W - Weightage (%) , T - Teaching hours**

| **Sr. No.** | **Topics** | **Weightage (%)** | **Teaching Hours** |
|---|---|---|---|
| 1 | **Introduction:**<br>What is DevOps? Why DevOps? Where DevOps is Useful? History of DevOps, How is DevOps different from traditional IT? Why is DevOps used? DevOps Workflow, How is DevOps different from Agile? DevOps Vs Agile, DevOps Principles, DevOps and Software Development Life Cycle, Roles, Responsibilities, and Skills of a DevOps Engineer, DevOps Automation Tools, What is the future of DevOps? | 15% | 7 |
| 2 | **Introduction to DevOps Automation Tools:**<br>Infrastructure Automation: Amazon Web Services (AWS), Configuration Management: Chef, Puppet, Deployment Automation: Jenkins, Log Management: Splunk, | 15% | 7 |

| | | | |
|---|---|---|---|
| | Performance Management: App Dynamic, Monitoring: Nagios | | |
| 3 | **Introduction to Docker:**<br>Docker Containers, Use of Docker, Virtualization vs. Docker, Benefits of Docker, Docker Architecture, Docker Engine, Docker Architecture in detail, Docker Installation, Provisioning | 15% | 7 |
| 4 | **Docker Cluster:**<br>Swarm Overview, Swarm Prerequisites, Create Swarm, Adding node to the Swarm, Swarm Deploy and Inspect Service, Swarm Delete Service, Swarm Drain | 15% | 7 |
| 5 | **Introduction to Jenkins:**<br>Jenkins Introduction, Build Cycle, Java GIT Installations, Obtaining and Installing Jenkins | 10% | 3 |
| 6 | **Automated Testing:**<br>Automated Testing, Automated Testing Jenkins Installation on Windows, Automation Testing Eclipse Kepler Installation, Automated Testing TestNG Installation, Automated Testing Selenium, Automation Testing Creating Java Project, Automated Creating and Testing Java Program, Automation Testing Creating Testing XML, Automation Testing Running TestNG XML, Automation Testing Creating Batch Script, Automation Testing Configuring Jenkins Job | 15% | 7 |
| 7 | **GIT:**<br>Version Control - GIT, GIT Features, 3-Tree Architecture, GIT Workflow, GIT Soft & Hard Reset, GIT Clone/Commit/Push, GIT Hub Projects, GIT Hub Management, GIT Rebase & Merge, GIT Stash, Reset, Checkout, GIT Clone, Fetch, Pull, GIT Branching Strategy | 15% | 7 |

**i. Reference Books:**


1. **DevOps for Beginners: Hands-on Guide**By David Johnson, 2016 edition. (TextBook)
2. **Building a DevOps Culture**
   By Mandi Walls, O'Reilly publications, 2013.
3. **The DevOps 2.0 Toolkit**
   By Viktor Farcic, 2016.

4. **Achieving DevOps**
   By Dave Harrison, Knox Lively, Apress publications, 2019.

**j. List of Practicals:**

1. Understand DevOps concepts, workflow, and its differences from traditional IT:
   - Define DevOps and explain its importance.
   - Describe the DevOps workflow.
   - Compare DevOps with Agile and traditional IT.
2. Set up AWS infrastructure and deploy a virtual machine:
   - Create an AWS account.
   - Launch an EC2 instance and connect via SSH.
   - Deploy a simple web application.
3. Automate configuration management using Chef or Puppet:
   - Install Chef or Puppet on a virtual machine.
   - Configure a node and apply a configuration script.
   - Automate the deployment of a web server.
4. Set up Jenkins and automate deployment:
   - Install Jenkins on Windows/Linux.
   - Configure a Jenkins job to pull code from GitHub.
   - Automate a build and deployment pipeline.
5. Implement log management and monitoring:
   - Install and configure Splunk for log analysis.
   - Set up Nagios for system monitoring.
   - Create alerts based on log patterns.
6. Understand Docker and deploy applications in containers:
   - Install Docker and run a container.
   - Create a Dockerfile and build an image.
   - Deploy a multi-container application using Docker Compose.

7. Manage container clusters using Docker Swarm:
    - Initialize a Swarm cluster.
    - Add nodes to the cluster.
    - Deploy and inspect services.
8. Automate testing using Selenium and TestNG:
    - Install Selenium and TestNG.
    - Write a test script in Java.
    - Execute the test using TestNG.
9. Manage source code using Git:
    - Install Git and configure a repository.
    - Perform commit, push, and pull operations.
    - Work with branching, merging, and rebasing.

## Semester 7-1

a. **Course:** Information and Network Security

b. **Course Code:** 303105432

c. **Prerequisite:** Basic knowledge of computer networks, cryptography, operating systems, programming, data structures, and cybersecurity fundamentals is required.

d. **Course Objective:** This course introduces the fundamental principles of cryptography and its applications on the network security domain as well as software development domain. This subject covers various important topics concerning information security like symmetric and asymmetric cryptography, hashing, message and user authentication, digital signatures, key distribution and overview of the malware technologies. The subject also covers the applications of all of these in real life situations

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLO 1** | Explain basic concepts of security and apply classical encryption techniques. |

| | |
|---|---|
| **CLO 2** | Apply block cipher principles, DES, and multiple encryption modes (3DES). |
| **CLO 3** | Apply number theory concepts and AES techniques for secure data transmission. |
| **CLO 4** | Implement and analyze public-key cryptosystems like RSA and Diffie-Hellman. |
| **CLO 5** | Apply hash functions and message authentication codes to ensure data integrity. |
| **CLO 6** | Explain and implement key management, digital signatures, and advanced security mechanisms (firewalls, malware defenses). |

**f. Course Outcome:**

| **CO No.** | **Course Outcome Description** |
|---|---|
| **CO 1** | Explain basic concepts of security and apply classical encryption techniques. |
| **CO 2** | Apply block cipher principles, DES, and multiple encryption modes (3DES). |
| **CO 3** | Apply number theory concepts and AES techniques for secure data transmission. |
| **CO 4** | Implement and analyze public-key cryptosystems like RSA and Diffie-Hellman. |
| **CO 5** | Apply hash functions and message authentication codes to ensure data integrity. |
| **CO 6** | Explain and implement key management, digital signatures, and advanced security mechanisms (firewalls, malware defenses). |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | |
|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | **ESE** | **Total** |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | 0 | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

**Course Content W - Weightage (%) , T - Teaching hours**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|
| 1 | **Introduction**<br>Computer Security Concept, The OSI Security Architecture, Security Attacks, Security Services, Security Mechanism, A Model for Network Security. | 6 | 2 |
| 2 | **Classical Encryption Techniques**<br>Symmetric Cipher Model, Cryptanalysis, Cryptanalysis Attacks, Substitution Techniques: Caesar Cipher, Monoalphabetic Cipher, Hill Cipher, Play fair Cipher, Polyalphabetic Cipher, OTP, Transposition Techniques, Steganography. | 12 | 6 |
| 3 | **Block Ciphers and the Data Encryption Standard**<br>Stream ciphers and block ciphers, Block Cipher Principles, Data Stream ciphers and block ciphers, Confusion & Diffusion, Block Cipher Principles, Data Encryption Standard (DES), Deferential and Linear Cryptanalysis, Avalanche Effect, strength of DES, Design principles of block cipher.<br>Multiple Encryption and Triple DES<br>Multiple encryption and triple DES, Electronic Code Book, Cipher Block Chaining Mode, Cipher Feedback mode, Output Feedback mode, Counter mode. | 20 | 14 |

| | | | |
|---|---|---|---|
| 4 | **Number theory** and Advance Encryption Standard<br>The Euclidean Algorithm, Modular Arithmetic, Finite Fields of the Form GF (p), Polynomial Arithmetic, Advance Encryption Standard (AES): structure, key expansion. | 15 | 6 |
| 5 | **Asymmetric Ciphers**<br>Prime Numbers, Principles of Public-Key Cryptosystems, The RSA Algorithm, Diffie-Hellman Key Exchange, Man in the Middle attack. | 15 | 4 |
| 6 | **Cryptographic Data Integrity Algorithms**<br>Hash Function: Hash Function and its Application, Security Requirements for Cryptographic Hash Functions, Hash Functions Based on Cipher Block Chaining, Secure Hash Algorithm (SHA), and MAC: Message Authentication Requirements, Message Authentication Functions, Requirements for Message Authentication Codes, Security of MACs, HMAC, Digital Signature: Introduction to Digital Signatures, Digital Signature Standard. | 20 | 8 |
| 7 | **Key Management and Distribution**<br>Symmetric Key Distribution: Symmetric Key Distribution Using Symmetric Encryption, Symmetric Key Distribution Using Asymmetric Encryption, Asymmetric Key Distribution: Distribution of Public Keys, X.509 certificates, Advanced Topics: Firewall, Intruders, Virus, Trojans, Malware, and Ransomware. | 12 | 5 |

**i. Reference Books:**


1. **Cryptographyand Network Security**
   By William Stallings | Pearson Education
2. **Cryptography& Network Security**
   By Behrouz A. Forouzan | Tata McGraw-Hill
3. **Information Security Principles and Practice**
   By Deven Shah, | Wiley-India
4. **Information Security Principles and Practice**
   By Mark Stamp, Willy India Edition
5. **Information systems security**
   By Nina Godbole | Wiley Publications,2008


**j. List of Practicals:**

**Practical-1:** Implement Caesar cipher encryption-decryption.

**Practical-2:** Implement Monoalphabetic cipher encryption-decryption.

**Practical-3:** Implement Playfair cipher encryption-decryption.
**Practical-4:** Implement Polyalphabetic cipher encryption-decryption.
**Practical-5:** Implement Hill cipher encryption-decryption.
**Practical-6:** Implement Simple Transposition encryption-decryption.
**Practical-7:** Implement One time pad encryption-decryption.
**Practical-8:** Implement Diffie-Hellman Key exchange Method.
**Practical-9:** Implement RSA encryption-decryption algorithm.
**Practical-10:** Demonstrate working of Digital Signature using Cryptool.

## Semester 7-2

a. **Course:** Data Science

b. **Course Code:** 303105432

c. **Prerequisite:** Basic knowledge of programming

d. **Rationale :** This course provides necessary knowledge on data manipulation and to perform analysis on the practical problems using statistical and machine learning approach to generate report and visualize the results in graphical form using programming tool.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLO 1** | Ability to gain basic knowledge on data science |
| **CLO 2** | Convert the real time data into suitable form for analysis |
| **CLO 3** | Gain the insights from the data through statistical inferences |
| **CLO 4** | Develop suitable models using machine learning techniques and to analyse its performance |
| **CLO 5** | Identify the requirement and visualize the results |
| **CLO 6** | Analyze on the performance of the model and the quality of the results |

**f. Course Outcome:**

| | |
|---|---|
| **CLO 1** | Ability to gain basic knowledge on data science |
| **CLO 2** | Convert the real time data into suitable form for analysis |
| **CLO 3** | Gain the insights from the data through statistical inferences |
| **CLO 4** | Develop suitable models using machine learning techniques and to analyse its performance |
| **CLO 5** | Identify the requirement and visualize the results |
| **CLO 6** | Analyze on the performance of the model and the quality of the results |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | 0 | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

**Course Content W - Weightage (%) , T - Teaching hours**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|
| 1 | **INTRODUCTION:** Data Science: Introduction to Data Science – Digital Universe – Sources of Data – Information Commons – Data Science Project Life, Cycle: OSEMN Framework | 9 | 4 |
| 2 | **DATA PREPROCESSING:** Introduction to Data Preprocessing – Reading, Selecting, Filtering Data – Filtering Missing Values – Manipulating, Sorting, Grouping, Rearranging, Ranking Data | 14 | 6 |
| 3 | **CONCEPT LEARNING:** Formulation of Hypothesis – Probabilistic Approximately Correct Learning - VC Dimension – Hypothesis elimination – Candidate, Elimination Algorithm | 16 | 6 |
| 4 | **BASICS OF R:** R Basics - data types and objects - control structures – data frame - Feature Engineering - scaling, Label Encoding and One Hot Encoding, Reduction | 19 | 8 |
| 5 | **MODEL FITTING IN R:** Regression Models- Linear and Logistic Model, Classification Models– Decision Tree, Naïve Bayes, SVM and Random Forest, Clustering Models – K Means and Hierarchical clustering | 19 | 9 |
| 6 | **VISUALIZATION:** Data visualization: Box plot, histogram, scatter plot, heat map – Working with Tableau – Outlier detection – Data Balancing | 14 | 7 |
| 7 | **PERFORMANCEEVALUATION in R:** Loss Function and Error: Mean Squared Error, Root Mean Squared Error – Model Selection and Evaluation criteria: Accuracy, Precision, F1 score, Recall Score – Binary Predictive Classification – Sensitivity, – Specificity. | 9 | 5 |

**i. Reference Books:**

1. Ethem Alpaydin, Introduction to Machine Learning, Fourth Edition, MIT Press, 2020 (Text Book)

2. Hadley Wickham, Garrett Grolemund, R for data science : Import, Tidy, Transform, Visualize, And Model Data Paperback, 2017
3. Han, J., Kamber, M., Pei, J. Data mining concepts and techniques. Morgan Kaufmann. 2011
4. James, G., Witten, D., T., Tibshirani, R. An Introduction to statistical learning with applications in R. Springer. 2013

**j. List of Practicals:**

1. House rent prediction using linear regression
2. Medical diagnosis for disease spread pattern
3. Automate email classification and response
4. Customer segmentation in business model based on their demographic, psycho graphic and behavior data
5. Construct a recommendation system based on the customer transaction using Association rule mining
6. Behavioral analysis of customers for any online purchase model
7. Agricultural data analysis for yield prediction and crop selection on Indian terrain data set
8. Develop a business model to predict the trend in Investment and Funding

**Semester 7-3**

**a. Course:** Cyber Physical Systems

**b. Course Code:** 303105450

**c. Prerequisite:** : Fundamentals of control systems, Basic knowledge of programming

**d. Rationale :** Cyber Physical Systems is a comprehensive understanding of the integration between computational algorithms and physical components in modern engineering systems. It emphasizes modeling, analysis, design, and implementation of CPS in real-world applications like autonomous vehicles, smart grids, and industrial automation.

**e. Course Learning Objectives:**

| | |
|---|---|
| **CLO 1** | Describe IoT fundamentals, embedded systems, M2M, and cloud/fog computing concepts. |

| | |
|---|---|
| **CLO 2** | Explain IoT architectures, the IoT technology stack, and the role of devices, sensors, actuators, and communication protocols. |
| **CLO 3** | Describe IoT communication protocols and standards, including Zigbee, Bluetooth, RFID, MQTT, CoAP, and related IoT networking technologies. |
| **CLO 4** | Demonstrate IoT hardware and software integration using Arduino, ESP8266, and Raspberry Pi. |
| **CLO 5** | Examine IoT applications, case studies, and open challenges, including security, scalability, mobility, interoperability, energy efficiency, and data management in IoT systems. |

**f. Course Outcome:**

| | |
|---|---|
| **CLO 1** | Describe IoT fundamentals, embedded systems, M2M, and cloud/fog computing concepts. |
| **CLO 2** | Explain IoT architectures, the IoT technology stack, and the role of devices, sensors, actuators, and communication protocols. |
| **CLO 3** | Describe IoT communication protocols and standards, including Zigbee, Bluetooth, RFID, MQTT, CoAP, and related IoT networking technologies. |
| **CLO 4** | Demonstrate IoT hardware and software integration using Arduino, ESP8266, and Raspberry Pi. |
| **CLO 5** | Examine IoT applications, case studies, and open challenges, including security, scalability, mobility, interoperability, energy efficiency, and data management in IoT systems. |

**g. Teaching & Examination Scheme:**

| **Teaching Scheme** | **Evaluation Scheme** |
|---|---|

| L | T | P | C | Internal Evaluation | | | ESE | | Total |
|---|---|---|---|---|---|---|---|---|---|
| | | | | MSE | CE | P | Theory | P | |
| 3 | 0 | - | 3 | 20 | 20 | - | 60 | - | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

**Course Content W - Weightage (%) , T - Teaching hours**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|
| 1 | **INTRODUCTION:** Data Science: Introduction to Data Science – Digital Universe – Sources of Data – Information Commons – Data Science Project Life, Cycle: OSEMN Framework | **9** | **4** |
| 2 | **DATA PREPROCESSING:** Introduction to Data Preprocessing – Reading, Selecting, Filtering Data – Filtering Missing Values – Manipulating, Sorting, Grouping, Rearranging, Ranking Data | **14** | **6** |
| 3 | **CONCEPT LEARNING:** Formulation of Hypothesis – Probabilistic Approximately Correct Learning - VC Dimension – Hypothesis elimination – Candidate, Elimination Algorithm | **16** | **7** |
| 4 | **BASICS OF R:** R Basics - data types and objects - control structures – data frame - Feature Engineering - scaling, Label Encoding and One Hot Encoding, Reduction | **19** | **8** |
| 5 | **MODEL FITTING IN R:** Regression Models- Linear and Logistic Model, Classification Models– Decision Tree, Naïve Bayes, SVM and Random Forest, Clustering Models – K Means and Hierarchical clustering | **19** | **9** |

| | | | |
|---|---|---|---|
| 6 | **VISUALIZATION:**<br>Data visualization: Box plot, histogram, scatter plot, heat map – Working with Tableau – Outlier detection – Data Balancing | **14** | **7** |
| 7 | **PERFORMANCEEVALUATION in R:**<br>Loss Function and Error: Mean Squared Error, Root Mean Squared Error – Model Selection and Evaluation criteria: Accuracy, Precision, F1 score, Recall Score – Binary Predictive Classification – Sensitivity, – Specificity. | **9** | **5** |

**i. Reference Books:**


1. **"Internet of Things (A Hands-on-Approach)",**

   By Vijay Madisetti and Arshdeep Bahga, | VPT

2. **Designing the Internet of Things**

   By Adrian McEwen, Hakim Cassimall | Wiley | 1st

3. **Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems**

   By Dr. Ovidiu Vermesan, Dr. Peter Friess | River Publishers

4. **Interconnecting Smart Objects with IP: The Next Internet**

   By Jean-Philippe Vasseur, Adam Dunkels | Morgan Kuffmann Publishers

5. **Getting Started with the Internet of Things**

   By Cuno Pfister, | O"Reilly Media


## Semester 7-4

**a. Course:** Real Time Systems

**b. Course Code:** 303105449

**c. Prerequisite**: Knowledge of basic computer hardware and OS

**d. Rationale:** This is a course on the design and applications of all real time aspects of various system components, like OS, memory, communication and an introduction to reliability evaluation methods

**e. Course Learning Objectives:**

| | |
|---|---|
| **CLO 1** | Explain RTOS concepts, scheduling, multitasking, and resource sharing. |

| | |
|---|---|
| **CLO 2** | Manage tasks with FreeRTOS features (creation, priorities, states). |
| **CLO 3** | Apply semaphores and event flags for synchronization. |
| **CLO 4** | Implement queues for inter-task communication. |
| **CLO 5** | Analyze RTOS internals (interrupts, timers, context switching). |
| **CLO 6** | Apply RTOS features like delays, suspension, and memory management. |

**f. Course Outcome:**

| | |
|---|---|
| **CO 1** | Explain RTOS concepts, scheduling, multitasking, and resource sharing. |
| **CO 2** | Manage tasks with FreeRTOS features (creation, priorities, states). |
| **CO 3** | Apply semaphores and event flags for synchronization. |
| **CO 4** | Implement queues for inter-task communication. |
| **CO 5** | Analyze RTOS internals (interrupts, timers, context switching). |
| **CO 6** | Apply RTOS features like delays, suspension, and memory management. |

**g. Teaching & Examination Scheme:**

| Teaching Scheme | | | | Evaluation Scheme | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| L | T | P | C | Internal Evaluation | | | ESE | | Total |
| | | | | MSE | CE | P | Theory | P | |
| 3 | 0 | 0 | 3 | 20 | 20 | - | 60 | - | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination

**h. Course Content:**

**Course Content W - Weightage (%) , T - Teaching hours**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|
| 1 | Introduction:<br>OS, RTOS, RTAs, Task scheduling, Latency, Multitasking, Priority Inversion & Priority Inheritance, OS Basics, Context Switching, Scheduling Resource sharing across Tasks: Resource sharing, Mutex, Recursive Mutex and Deadlocks | 16 | 7 |
| 2 | Task Management:<br>Task Creation, Starting the scheduler for scheduling of tasks, Task control block, IDLE task, FreeRTOS Timer services task, FreeRTOS Scheduler, FreeRTOS Task Notifications, Task Deletion, FreeRTOS Task Priority, FreeRTOS Task states | 20 | 9 |
| 3 | Semaphores:<br>Synchronization and Mutual Exclusion, Semaphores, Counting and Binary Semaphores, Mutex Vs Semaphores, Semaphore APIs, Events Latching<br>Event Flag Groups: Introduction, APIs | 16 | 7 |

| | | | |
|---|---|---|---|
| 4 | Queue Management:<br>Message Queues, Message Queue APIs, Queue Creation, Sending and Receiving data from Queue | 16 | 7 |
| 5 | RTOS Internals:<br>Macros, variables, Functions, Interrupts, Timers, Ticks, FreeRTOS Hook Functions, Context switching points, Ready list, Stack overflow | 16 | 7 |
| 6 | Additional RTOS features:<br>Task Delay, Critical session, Task suspend, Memory management, Heap and stack management, FreeRTOS Synchronization and Mutual Exclusion Services, FreeRTOS Delay APIs | 16 | 7 |

**i. Reference Books:**


1. Mastering the FreeRTOS Real-Time Kernel: A Hands-On Tutorial Guide By Richard Barry (TextB
2. https://www.udemy.com/course/mastering-rtos-hands-on-with-freertos-arduino-and-stm32f Kiran Nayak | FastBit Embedded Brain Academy
3. Hands-On RTOS with Microcontrollers: Building real-time embedded systems using FreeRTO MCUs, and SEGGER debug tools by Brian Amos | 1st Edn., Packt
4. Real-time Operating Systems: Book 1 - The Theory (The engineering of real-time embedded sy Jim Cooling
5. Real-time Operating Systems Book 2 - The Practice: Using STM Cube, FreeRTOS and the STM32 Board: 1 (Engineering of Real-Time Embedded Systems) By Jim Cooling
6. http://www.freertos.org/a00106.html


## Semester 7-5

**a. Course Name:** Cyber Security

**b. Course Code:** 303105341

**c. Prerequisite:** Fundamental of Programming, Computer Network

**d. Rationale:** Cyber security is the application of technologies, processes, and controls to protect systems, networks, programs, devices, and data from cyber-attacks. It aims to reduce the risk of cyber-attacks and protect against the unauthorized exploitation of systems, networks, and technologies.

**e. Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Explain the features and characteristics of the Linux Operating System and Windows Operating System. |
| **CLOBJ 2** | Apply network monitoring tools to identify attacks against network protocols and services. |
| **CLOBJ 3** | Apply various methods to prevent malicious access to computer networks, hosts, and data. |
| **CLOBJ 4** | Explain how to investigate endpoint vulnerabilities and attacks. |
| **CLOBJ 5** | Analyze network intrusion data to verify potential exploits. |
| **CLOBJ 6** | Apply incident response models to manage network security incidents |

**f. Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Explain the features and characteristics of the Linux Operating System and Windows Operating System. |
| **CLO 2** | Apply network monitoring tools to identify attacks against network protocols and services. |
| **CLO 3** | Apply various methods to prevent malicious access to computer networks, hosts, and data. |
| **CLO 4** | Explain how to investigate endpoint vulnerabilities and attacks. |

| | |
|---|---|
| **CLO 5** | Analyze network intrusion data to verify potential exploits. |
| **CLO 6** | Apply incident response models to manage network security incidents |

**Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | 0 | 2 | 4 | 20 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination **Course Content:**

| **Sr. No.** | **Topics** | **Weightage (%)** | **Teaching Hours** |
|---|---|---|---|
| 1 | **Information Security:** Introduction to information systems, Types of information Systems, Development of Information Systems, Introduction to Information Security, Need for Information Security, Threats to Information Systems, Information Assurance, Cyber Security and Security Risk Analysis. Emerging trends in Information Security, Role of AI in cybersecurity, and Zero Trust Architecture. | **15** | **8** |
| 2 | **Systems Vulnerability Scanning:** Overview of vulnerability scanning, Open Port/Service Identification, Banner/ Version Check, Traffic Probe, Vulnerability Probe, Vulnerability Examples. Networks Vulnerability Scanning - Netcat, Understanding Port and Services tools, Network Reconnaissance–Nmap. Network Sniffers and Injection tools–Wireshark**. Advanced Persistent Threat (APT) detectionand Automated** | **25** | **13** |

| | | | |
|---|---|---|---|
| | **Vulnerability Assessment tools.** | | |
| 3 | **Firewalls and Packet Filters:** Firewall Basics, Packet Filter Vs Firewall, Firewall Protects a Network, Packet Characteristic to Filter, Stateless Vs Stateful Firewalls, Network Address Translation (NAT) and Port Forwarding, the basics of Virtual Private Networks, Linux Firewall, Windows Firewall, Snort: Introduction Detection System. **Role of Intrusion Prevention Systems (IPS) and Next-Generation Firewalls (NGFW)** | **20** | **11** |
| 4 | **Introduction to Cyber Crime and Law:** Cyber Crimes, Types of Cybercrime, Hacking, Attack vectors, Cyberspace and Criminal Behavior, Clarification of Terms, Traditional Problems Associated with Computer Crime, Introduction to Incident Response, Digital Forensics, Computer Language, Network Language, Realms of the Cyber world, A Brief History of the Internet, Recognizing and Defining Computer Crime, Contemporary Crimes, Computers as Targets, Contaminants and Destruction of Data, Indian ITACT 2000. **Latest amendments in cyber laws, GDPR, and ethical hacking regulations.** | **20** | **7** |
| 5 | **Introduction to Cyber Crime Investigation**: Firewalls and Packet Filters, password Cracking, Key loggers and Spyware, Virus and Worms, Trojan and backdoors, Steganography, DOS and DDOS attack, SQL injection, Buffer Overflow, Attack on wireless Networks. **Deepfake detection, Social Engineering Tactics, and AI-driven Cyber Threat Intelligence.** | **20** | **7** |

**Reference Books:**


1. Cryptographyand Network Security (TextBook) By William Stallings | Pearson Education
2. Cryptographyand Network Security (TextBook) By V.K. Jain | Khanna Publishing House
3. Information and Cyber Security (TextBook) By Gupta Sarika| Khanna Publishing House
4. Cryptographyand Network Security (TextBook) By Atul Kahate | TMH
5. Cryptographyand Information Security (TextBook) By V.K. Pachghare | PHI Learning

6. Anti-Hacker Tool Kit By Mike Shema | McGrawHill
7. Cyber Security understanding Cyber Crimes, Computerforensics and Legal Perspectives By Nina Godbole and Sunit Belapure| WILEY

**Semester 7-6**

a. **Course Name:** Image Processing

b. **Course Code:** 303105381

c. **Prerequisite:** Knowledge of Fourier Transform and Digital Signal Processing

d. **Rationale:** This is fundamental course of computer vision. This course will strengthen fundamental knowledge about digital image processing techniques. Digital image processing is used in almost all engineering fields and wide range of applications in industrial automation, medical, agriculture, security, entertainment, education and many more.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Explain fundamentals of digital image processing, visual perception, sensing, sampling, and quantization. |
| **CLOBJ 2** | Apply spatial and frequency domain techniques for image enhancement and noise reduction. |
| **CLOBJ 3** | Implement image restoration techniques and filtering methods in spatial and frequency domains. |
| **CLOBJ 4** | Analyze and apply image compression models and standards (error-free and lossy). |
| **CLOBJ 5** | Apply morphological operations and segmentation methods for feature extraction and object recognition. |

| | |
|---|---|
| **CLOBJ 6** | Evaluate image processing models and design applications for real-world scenarios. |

**f. Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Explain fundamentals of digital image processing, visual perception, sensing, sampling, and quantization. |
| **CLO 2** | Apply spatial and frequency domain techniques for image enhancement and noise reduction. |
| **CLO 3** | Implement image restoration techniques and filtering methods in spatial and frequency domains. |
| **CLO 4** | Analyze and apply image compression models and standards (error-free and lossy). |
| **CLO 5** | Apply morphological operations and segmentation methods for feature extraction and object recognition. |
| **CLO 6** | Evaluate image processing models and design applications for real-world scenarios. |

**Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | 0 | 2 | 4 | 40 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-**

Continuous Evaluation; **ESE-** End Semester Examination **Course Content:**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|
| 1 | **Digital Image Fundamentals** What Is Digital Image Processing?, The Origins of Digital Image Processing, Examples of Fields that Use Digital Image Processing, Fundamental Steps in Digital Image Processing, Components of an Image Processing System. Elements of Visual Perception, Light and the Electromagnetic Spectrum, Image Sensing and Acquisition, Image Sampling and Quantization, Some Basic Relationships Between Pixels, Linear and Nonlinear Operations. | 12 | 5 |
| 2 | **Image enhancement and filtering in spatial domain** Background, Some Basic Gray Level Transformations, Histogram Processing, Enhancement Using Arithmetic/Logic Operations, Basics of Spatial Filtering, Smoothing Spatial Filters. | 16 | 7 |
| 3 | **Image filtering in the frequency domain** Background, Introduction to the Fourier Transform and the Frequency Domain, Smoothing Frequency-Domain Filters, Sharpening Frequency Domain Filters, Homomorphic Filtering. | 16 | 7 |
| 4 | **Image restoration** A Model of the Image Degradation/Restoration Process, Noise Models, Restoration in the Presence of Noise Only Spatial Filtering, Periodic Noise Reduction by Frequency Domain Filtering, Linear, Position- Invariant Degradations, Estimating the Degradation Function, Inverse Filtering, Minimum Mean Square Error (Wiener) Filtering | 16 | 7 |
| 5 | **Image Compression** Fundamentals, Image Compression Models, Elements of Information Theory, Error-Free Compression, Lossy Compression, Image Compression Standards. | 10 | 5 |

| | | | |
|---|---|---|---|
| 6 | **Morphological Image Processing** Preliminaries, Dilation and Erosion, Opening and Closing, The Hit-or-Miss Transformation, Some Basic Morphological Algorithms. | 10 | 5 |
| 7 | **Image Segmentation** Detection of Discontinuities, Edge Linking and Boundary Detection, Thresholding , Region- Based Segmentation | 10 | 5 |
| 8 | **Object Recognition** Patterns and Pattern Classes, Recognition Based on Decision-Theoretic Methods, Structural Methods | 10 | 4 |

**Reference Books:**


| | |
|---|---|
| 1. | **Digital Image Processing**<br>By Rafael C. Gonzalez and Richard E. Woods \| Pearson Education |
| 2. | **Fundamentals of Digital Image Processing**<br>By A. K. Jain \| Pearson Education |
| 3. | **Digital Image Processing**<br>By Pratt William \| John Wiley & Sons |
| 4. | **Digital Image Processing Using MATLAB**<br>By Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins \| Tata McGraw Hill |


**List of Practical**

1. To study and implement basic commands of MATLAB required for digital image processing techniques, and various image file formats.

2. To study the effect of down sampling and quantization techniques on the grayscale image.

3. Write MATLAB code to enhance the visual quality of the image using point processing techniques, namely a) Image negative, b) Logarithmic transformation, and c) Power law transformation and Gray level slicing technique.

4. Write MATLAB code to display the histogram of the image and enhance the visual quality of the image with the help of histogram equalization technique.

5. Write MATLAB code to perform the bit-plane slicing method on grayscale image.

6. Write MATLAB code to demonstrate that the convolution in spatial domain is equivalent to multiplication in the frequency domain.

7. Write MATLAB code to restore grayscale image from noisy image with the help of image restoration techniques.

8. Write MATLAB code to perform Min, Median & Max Filtering on grayscale image.

9. Write MATLAB code to blur the image using Ideal low pass, Butterworth low pass and Gaussian Low pass filter.

10. Write MATLAB code to blur the image using Ideal High pass, Butterworth High pass and Gaussian High pass filter.

## Semester 7-7

a. **Course Name: Big Data Analytics**

b. **Course Code:** 303105361

c. **Prerequisite:** Database Management system, SQL

d. **Rationale:** Big data analytics is the often complex process of examining big data to uncover information such as hidden patterns, correlations, market trends and customer preferences that can help organizations make informed business decisions.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Explain big data concepts, characteristics, challenges, and analytics approaches. |
| **CLOBJ 2** | Differentiate SQL, NoSQL, and NewSQL data models, and apply NoSQL techniques for data management. |
| **CLOBJ 3** | Implement Hadoop ecosystem components (HDFS, MapReduce) for large-scale data processing. |
| **CLOBJ 4** | Examine Hadoop workflows, job execution, input/output formats, and performance aspects. |
| **CLOBJ 5** | Utilize Hadoop-related tools (Hive, Pig, HBase, Cassandra) for data querying, analysis, and reporting. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Explain big data concepts, characteristics, challenges, and analytics approaches. |
| **CLO 2** | Differentiate SQL, NoSQL, and NewSQL data models, and apply NoSQL techniques for data management. |
| **CLO 3** | Implement Hadoop ecosystem components (HDFS, MapReduce) for large-scale data processing. |
| **CLO 4** | Examine Hadoop workflows, job execution, input/output formats, and performance aspects. |
| **CLO 5** | Utilize Hadoop-related tools (Hive, Pig, HBase, Cassandra) for data querying, analysis, and reporting. |

**Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | 0 | 2 | 4 | 40 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination **Course Content:**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|
| 1 | **Introduction:**<br><br>What is in Store? , Classification of Digital Data: Structured, Semi Structured & Un Structured , Evolution of Big Data , Definition of Big Data - Volume - Velocity ±Variety, Challenges of Big Data , Why Big Data?, Traditional Business Intelligence (BI) versus Big Data , industry examples of big data , What is Big Data Analytics? , Data Science | **20** | **9** |
| 2 | **Nosql Data Management:**<br><br>Introduction to NoSQL, Types of NoSQL, Why NoSQL? , Advantages of NoSQL, Comparison of SQL, NoSQL and NewSQL , aggregates , key-value and document data models, graph databases, map-reduce, partitioning and combining | **20** | **9** |
| 3 | **Basics Of Hadoop:**<br><br>What is Hadoop?, Brief History of Hadoop , Why Hadoop? , RDBMS versus Hadoop , Hadoop Components , High Level Architecture of Hadoop , Key Advantages & Features of Hadoop , Data format ,Hadoop distributed file system (HDFS) , Processing Data with Hadoop. Map Reduce Interface:Overview of Map Reduce, Map-Reduce workflows, anatomy of Map-Reduce job run, shuffle and sort ,task execution ,input formats , output formats. | **40** | **18** |
| 4 | **Hadoop Related Tools:**<br><br>Overview of HBase, Pig introduction, Pig data model, Hive, data types and file formats, HiveQL data definition, HiveQL data manipulation, HiveQL queries, Pig Latin Overview , Pig versus Hive, Using JSON , Overview of Cassandra, Jasper Reports. | **20** | **9** |

**Reference Books:**


| | |
|---|---|
| 1. | **Hadoop: The Definitive Guide by Tom White, Third Edition, O'Reilley. (TextBook)** |

| | |
|---|---|
| | By Tom White |
| 2. | **Understanding Bigdata**<br>By Chris Eaton,Dirk derooset al. \| McGraw Hill, Pub. Year 2012 |
| 3. | **Hadoop Operations**<br>By Eric Sammer \| O'Reilley, Pub. Year 2012 |
| 4. | **Big data analytics with R and Hadoop, VigneshPrajapati,SPD.**<br>By VigneshPrajapati |
| 5. | **Big Data and Analytics**<br>By Seema Acharya and Subhashini C \| Wiley India |
| 6. | **Programming Hive, E. Capriolo, D. Wampler, and J. Rutherglen, O'Reilley**<br>By E. Capriolo, D. Wampler, and J. Rutherglen |
| 7. | **MongoDB in Action**<br>By Kyle Banker, Piter Bakkum ,Shaun Verch \| Dream tech Press |

**List of Practical**

1. To study the basic concepts of Big Data Analytics and install/configure Hadoop ecosystem components such as Hadoop and HDFS.

2. Write commands to perform basic HDFS operations such as creating directories, uploading files, listing files, downloading files, and deleting files in Hadoop Distributed File System.

3. Write a MapReduce program to perform Word Count on a large text dataset using Hadoop.

4. Write a MapReduce program to find the maximum, minimum, and average values from a given dataset.

5. Write a MapReduce program to sort a large dataset using Hadoop framework.

6. Write Hive queries to create, load, and query tables in Hive for analyzing large datasets.

7. Write Hive queries to perform aggregation functions, filtering, and joins on large datasets.

8. Write Pig Latin scripts to perform data analysis operations such as filtering, grouping, and sorting on large datasets.

9. Write a Spark program to perform data processing operations such as word count and data transformation using Apache Spark.

10. Write a Spark program to analyze a large dataset and visualize basic analytics results such as counts, averages, or trends.

## Semester 7-8

a. **Course Name:** High Performance Computing

b. **Course Code:** 303105355

c. **Prerequisite:** Parallel Computing, Computer Architecture

d. **Rationale:** HPC is more than just for achieving high performance - it is a compelling vision for how computation can seamlessly scale from a single processor to virtually limitless computing power. The market demands general-purpose processors that deliver high single threaded performance as well as multi-core throughput for a wide variety of workloads on client, server, and high- performance computing (HPC) systems.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Explain parallel computing fundamentals, levels of parallelism, computation models, and modern architectures like multi-core and multi-threaded systems. |
| **CLOBJ 2** | Apply principles of parallel algorithm design, including decomposition, task mapping, and interaction management. |
| **CLOBJ 3** | Implement MPI-based message-passing for communication and computation in parallel programs |
| **CLOBJ 4** | Utilize thread-based parallelism and synchronization using Pthreads and OpenMP for efficient task execution. |
| **CLOBJ 5** | Apply GPGPU architectures and CUDA programming to design and implement parallel applications. |

| | |
|---|---|
| **CLOBJ 6** | Analyze and evaluate parallel program performance using metrics like speedup, efficiency, scalability, and apply Amdahl's Law. |

**f. Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Explain parallel computing fundamentals, levels of parallelism, computation models, and modern architectures like multi-core and multi-threaded systems. |
| **CLO 2** | Apply principles of parallel algorithm design, including decomposition, task mapping, and interaction management. |
| **CLO 3** | Implement MPI-based message-passing for communication and computation in parallel programs |
| **CLO 4** | Utilize thread-based parallelism and synchronization using Pthreads and OpenMP for efficient task execution. |
| **CLO 5** | Apply GPGPU architectures and CUDA programming to design and implement parallel applications. |
| **CLO 6** | Analyze and evaluate parallel program performance using metrics like speedup, efficiency, scalability, and apply Amdahl's Law. |

**Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | 0 | 2 | 4 | 40 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination **Course Content:**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|
| 1 | **Digital Image Fundamentals** What Is Digital Image Processing ?, The Origins of Digital Image Processing, Examples of Fields that Use Digital Image Processing, Fundamental Steps in Digital Image Processing, Components of an Image Processing System. Elements of Visual Perception, Light and the Electromagnetic Spectrum, Image Sensing and Acquisition, Image Sampling and Quantization, Some Basic Relationships Between Pixels, Linear and Nonlinear Operations. | 12 | 5 |
| 2 | **Image enhancement and filtering in spatial domain** Background, Some Basic Gray Level Transformations, Histogram Processing, Enhancement Using Arithmetic/Logic Operations, Basics of Spatial Filtering, Smoothing Spatial Filters. | 16 | 7 |
| 3 | **Image filtering in the frequency domain** Background, Introduction to the Fourier Transform and the Frequency Domain, Smoothing Frequency-Domain Filters, Sharpening Frequency Domain Filters, Homomorphic Filtering. | 16 | 7 |
| 4 | **Image restoration** A Model of the Image Degradation/Restoration Process, Noise Models, Restoration in the Presence of Noise Only Spatial Filtering, Periodic Noise Reduction by Frequency Domain Filtering, Linear, Position- Invariant Degradations, Estimating the Degradation Function, Inverse Filtering, Minimum Mean Square Error (Wiener) Filtering | 16 | 7 |
| 5 | **Image Compression** Fundamentals, Image Compression Models, Elements of Information Theory, Error-Free | 10 | 5 |

| | | | |
|---|---|---|---|
| | Compression, Lossy Compression, Image Compression Standards. | | |
| **6** | **Morphological Image Processing** Preliminaries, Dilation and Erosion, Opening and Closing, The Hit-or-Miss Transformation, Some Basic Morphological Algorithms. | **10** | **5** |

**Reference Books:**


| | |
|---|---|
| 1. | Introduction to Parallel Computing By Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar \| Pearson Publication (TextBook) |
| 2. | Advanced Computer Architecture: Parallelism, Scalability, Programmability By Kai Hwang, Naresh Jotwani, McGraw Hill,Second Edition, 2010 |
| 3. | CUDA by Example An Introduction to General Purpose GPU Programming By Edward Kandrot and Jason Sanders, Addison- Wesley Professional, 2010 |


**List of Practical**

**1.** To study the basic concepts of High Performance Computing and implement simple C programs to measure execution time of sequential programs.

**2.** Write a program using OpenMP to demonstrate parallel execution of a simple loop and compare its execution time with sequential execution.

**3.** Write a program using OpenMP to implement parallel matrix addition and analyze the performance improvement.

**4.** Write a program using OpenMP to perform parallel matrix multiplication and compare results with the sequential implementation.

**5.** Write a program using OpenMP to demonstrate parallel reduction operation for calculating the sum of elements in an array.

**6.** Write a program using MPI to demonstrate communication between two processes using point-to-point communication (send and receive).

**7.** Write a program using MPI to implement parallel vector addition using multiple processes.

**8.** Write a program using MPI to implement parallel matrix multiplication and analyze its performance.

**9.** Write a program to demonstrate parallel sorting algorithms such as parallel bubble sort or parallel merge sort using OpenMP or MPI.

**10.** Write a program to measure and analyze speedup and efficiency of parallel programs using different numbers of processors.

## Semester 7-9

**g. Course Name:** Augmented and Virtual Reality

**h. Course Code:** 303105485

**i. Prerequisite:** Basic knowledge of programming (C#/C++/Python), computer graphics, 3D modeling and line.

**j. Rationale:** Develop, and implement immersive Augmented and Virtual Reality applications.

**k. Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Explain the fundamentals of virtual reality, including its history, paradigms, system components, and methods of user interaction. |
| **CLOBJ 2** | Apply geometric and transformation models to represent and manipulate objects and viewpoints in virtual environments. |
| **CLOBJ 3** | Analyze motion in real and virtual worlds, including velocities, accelerations, vestibular effects, and physics-based interactions. |
| **CLOBJ 4** | Evaluate VR applications, identify promising application fields, and design VR solutions based on usability and interaction principles. |
| **CLOBJ 5** | Explain augmented reality concepts, marker detection techniques, pose calculation, and tracking stability, and assess suitable AR application scenarios. |

| | |
|---|---|
| **CLOBJ 6** | Develop AR applications by implementing user interfaces, authoring dynamic content, and applying practical AR design and deployment considerations. |

**l. Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Explain the fundamentals of virtual reality, including its history, paradigms, system components, and methods of user interaction. |
| **CLO 2** | Apply geometric and transformation models to represent and manipulate objects and viewpoints in virtual environments. |
| **CLO 3** | Analyze motion in real and virtual worlds, including velocities, accelerations, vestibular effects, and physics-based interactions. |
| **CLO 4** | Evaluate VR applications, identify promising application fields, and design VR solutions based on usability and interaction principles. |
| **CLO 5** | Explain augmented reality concepts, marker detection techniques, pose calculation, and tracking stability, and assess suitable AR application scenarios. |
| **CLO 6** | Develop AR applications by implementing user interfaces, authoring dynamic content, and applying practical AR design and deployment considerations. |

**Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 3 | 0 | 2 | 4 | 40 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination **Course Content:**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|
| **1** | **Introduction of Virtual Reality and Augmented Reality:**<br><br>Introduction to Augmented-Virtual and mixed reality:A Brief History of Virtual Reality Introduction, Features of Augmented Reality, Difference between AR VR and MR, Challenges with AR, AR System and Functionality, AR methods, Visualization techniques for AR. The five Classic Components of a VR System, Primary Features and Present Development on Virtual Reality. | **15** | **8** |
| **2** | **Virtual Reality Systems:**<br><br>VR as a discipline, Basic features of VR Systems, Architecture of VR Systems. VR Hardware: VR input Hardware: Tracking systems, motion capture systems, Data Gloves, VR Output Hardware: Visual Display | **10** | **6** |
| **3** | **Geometric Modelling and Transformation:**<br><br>Geometric Modelling: Introduction, From 2D to 3D, 3D space curves, 3D boundary representation. Geometrical Transformations: Introduction, Frames of reference, Modelling transformations, Instances, Picking, Flying, Scaling the VE, Collision detection. Generic VR system: Introduction, Virtual environment, Computer environment, VR technology, Model of interaction, VR Systems. | **15** | **8** |
| **4** | **VR Hardware and Software:**<br><br>Human factors: Introduction, the eye, the ear, the somatic senses. VR Hardware: Introduction, sensor hardware, Head- | **25** | **8** |

| | | | |
|---|---|---|---|
| | coupled displays, Acoustic hardware, Integrated VR systems. VR Software: Challenges to VR Software Development, Master/Slave and Client/Server Architecture, Cluster Rendering, Game Engines and available SDK to develop VR Applications for different Hardware(HTC VIVE, Oculus, Google VR), Standard; Vega, MultiGen, Virtools etc.Introduction to VRML. | | |
| **5** | **3D Interaction Techniques:**<br><br>3D manipulation tasks, manipulation techniques and input devices, interaction techniques for 3D manipulation. | **10** | **5** |
| **6** | **AR Software Development and Applications:**<br><br>AR Software, Camera Parameters and Camera Calibration, Marker based AR, AR Toolkit, AR Devices, Smartphone, AR Glasses(Hollow Lens, Magic Leap, Head-Mounted Display (HMD), Sensor and Cameras). | **10** | **5** |
| **7** | **VR Applications:**<br><br>VR in Engineering, Science, Medical, Military, Training, Digital Entertainment: VR technology in Film & TV Production. VR Technology in Physical Exercises and Games. Demonstration of Digital Entertainment by VR. | **15** | **5** |

**Reference Books:**


| | |
|---|---|
| 1. | Burdea, G. C. and P. Coffet. Virtual Reality Technology, Second Edition. Wiley-IEEE Press, 2003/2006 (TextBook) |
| 2. | Alan B. Craig, Understanding Augmented Reality, Concepts and Applications, Morgan Kaufmann, 2013. |
| 3. | Alan Craig, William Sherman and Jeffrey Will, Developing Virtual Reality Applications, Foundations of Effective Design, Morgan Kaufmann, 2009. |


**List of Practical**

1. To study the basic concepts of Augmented Reality (AR) and Virtual Reality (VR) and explore different AR/VR development tools and platforms such as Unity and Vuforia.

2. To install and configure Unity 3D and set up an AR/VR development environment.
3. Create a simple 3D environment in Unity and implement basic navigation and interaction with objects.
4. Develop a basic Augmented Reality application to display a 3D object on a predefined image marker using Vuforia in Unity.
5. Create an AR application to overlay text, images, or 3D models on real-world objects using image tracking.
6. Develop a simple Virtual Reality scene in Unity and enable user interaction using VR controllers or keyboard inputs.
7. Create a VR application that allows users to move inside a virtual environment and interact with objects.
8. Develop an AR application to display animated 3D objects on a detected surface or marker.
9. Implement object manipulation techniques such as scaling, rotation, and translation of 3D objects in AR/VR environment.
10. Develop a simple AR/VR application integrating multiple interactive objects and user interactions in Unity.

**Semester 7-10**

**m. Course Name:** Block chain Technologies

**n. Course Code:** 303105383

**o. Prerequisite:** Introduction to Blockchain Technologies

**p. Rationale:** This course deals with understanding the concepts behind smart contracts which is the computational element of Blockchain Technology. It will cover Ethereum fundamentals and Solidity language programming basics for deployment of smart contracts. In this course, students will learn to design, code, deploy and execute smart contracts. Testing and deployment fo r smart contracts will be done using the Remix development environment. The course also covers concepts on how to interact with smart contracts as well as the security aspects.

**q. Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Explain the fundamentals, properties, applications, and components of blockchain, and differentiate between centralized and distributed ledger systems. |
| **CLOBJ 2** | Apply cryptographic primitives such as hashing, digital signatures, and Merkle trees to ensure security and trust in blockchain systems. |
| **CLOBJ 3** | Analyze distributed consensus mechanisms and evaluate their role in achieving reliability and fault tolerance in blockchain networks. |
| **CLOBJ 4** | Demonstrate the working of cryptocurrencies with emphasis on Bitcoin transactions, wallets, mining, and associated vulnerabilities. |
| **CLOBJ 5** | Design and evaluate smart contracts and compare blockchain platforms such as Ethereum, Hyperledger, Ripple, and R3 Corda for real-world applications. |
| **CLOBJ 6** | Compare different blockchain platforms (Ethereum, Hyperledger, IBM Blockchain, Ripple, R3 Corda, etc.) and assess their suitability for various applications. |

**r. Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Explain the fundamentals, properties, applications, and components of blockchain, and differentiate between centralized and distributed ledger systems. |
| **CLO 2** | Apply cryptographic primitives such as hashing, digital signatures, and Merkle trees to ensure security and trust in blockchain systems. |
| **CLO 3** | Analyze distributed consensus mechanisms and evaluate their role in achieving reliability and fault tolerance in blockchain networks. |
| **CLO 4** | Demonstrate the working of cryptocurrencies with emphasis on Bitcoin transactions, wallets, mining, and associated vulnerabilities. |
| **CLO 5** | Design and evaluate smart contracts and compare blockchain platforms such as Ethereum, Hyperledger, Ripple, and R3 Corda for real-world applications. |
| **CLO 6** | Compare different blockchain platforms (Ethereum, Hyperledger, IBM Blockchain, Ripple, R3 Corda, etc.) and assess their suitability for various applications. |

**Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | 0 | 2 | 4 | 40 | 20 | 20 | 60 | 30 | 150 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination **Course Content:**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|
| 1 | **UNIT-1**<br>**Introduction to Smart contracts and Ethereum Blockchain**<br>Smart contracts: History, Introduction, How Smart Contract Works, pros and cons of using smart contracts, use cases/ Applications of Blockchain and Smart Contracts<br>Ethereum Fundamentals: Ethereum, Adoption, Bitcoin vs Ethereum, Ether, Gas and fees, Ethereum Blockchain, Ethereum Nodes, ethereum accounts- externally owned, contract accounts, Ethereum transactions and blocks,<br>Signing Transactions, Mist wallet, MetaMask wallet | **20** | **8** |
| 2 | **UNIT-2**<br>**Introduction to Solidity programming basics:**<br>Structure of a contract: Data types and global variables, Access Modifiers, Data Structures - Arrays, Structs, Enum, Mappings, Expressions and control structures, Error Handling, Assert, Require, Revert, Function modifiers and fallbacks, Events, Structure of contract, ERC20 Token Interface, Development workflow. | **20** | **10** |
| 3 | **UNIT-3**<br>**.Object Oriented Concepts and Deployment:**<br>Polymorphism, Method overriding, Abstract Contract, Interfaces, Inheritance, Constructors, Deploying Smart Contracts. | **20** | **8** |
| 4 | **UNIT-4**<br>**Interacting with Smart contracts**:<br>Deployment Environment, Differences, Application development life cycle management, The Truffle Development Environment for web, Testing with Truffle, Working with Injected Web3, Working with Web3 Provider. | **20** | **10** |

| | | | |
|---|---|---|---|
| **5** | **UNIT-5**<br>**Securing Smart Contracts:**<br><br>Smart Contract, Errors, Smart contracts vulnerabilities, Attacks in Smart Contracts, Preventive Methodologies, DApp Safety, Proactive Controls, Security Measures | **20** | **6** |

**Reference Books:**


| | |
|---|---|
| 1. | Hands-On Smart Contract Development with Solidity and Ethereum |
| 2. | Ritesh Modi ,"Solidity Programming Essentials", Packt Publishing, April 2018 |
| 3. | Reed, Jeff (2016). Smart contracts: The essential guide to using blockchain smart contracts for cryptocurrenc y |


**List of Practical**

| | |
|---|---|
| **1.** | Introduction to ETHEREUM tools and Solidity |
| **2.** | Deploy a smart contract for printing "Hello World" using Java Script VM, Injected Web3 and Web3Provider using Metamask and Ganache. |
| **3.** | Deploy a smart contract for arithmetic operations using Java Script VM, Injected Web3 and Web3Provider using Metamask and Ganache. |
| **4.** | Deploy a smart contract for FINDING LARGEST NUMBER OUT OF THREE NUMBERS using Java Script VM, Injected Web3 and Web3Provider using Metamask and Ganache. |
| **5.** | Create a Smart Contract for a banking application in solidity which allows users to do the following: Mint money into your account Withdraw money from your account Send money from your account to smart contract address Check balance After a contract is created, deploy the contract on Ethereum Testnet network |
| **6.** | How to build a smart contract that lets user book rooms and pay for them with cryptocurrency |
| **7.** | Building a blockchain raffle using Solidity programming language. Apart from a coin toss, the most straightforward example of gambling is probably a raffle. |
| **8.** | Installation of Ether Wallet and deploying smart contract |
| **9.** | To design an electronic voting system, using the ethereum blockchain (smart contracts) and more precisely the RPC test which enables account generation with a private and public key. Blockchain electronic voting system using smart contracts. |
| **10.** | Building an improved P2P file system to provide originality and authenticity of published and posted free online digital content such as books, music, and movies. Our solution utilizes a blend of the latest emerging technologies that include IPFS and blockchain smart contracts. |

## Semester 7-11

a. **Course Name:** Computational Number Theory

b. **Course Code:** 303105451

c. **Prerequisite:** Students should have basic knowledge of Discrete Mathematics, basic algebra, and fundamental programming concepts.

d. **Rationale:** This course provides students with the theoretical foundations of number theory along with computational algorithms used for prime number generation, modular arithmetic, and cryptographic protocols. The knowledge gained from this course enables students to understand and implement secure encryption techniques and efficient numerical algorithms used in real-world applications.

e. **Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Define the fundamental concepts of number theory including divisibility, modular arithmetic, and congruences used in computational problems. |
| **CLOBJ 2** | Apply algorithms for prime number generation, primality testing, and integer factorization. |
| **CLOBJ 3** | Analyze and implement efficient algorithms for modular arithmetic and modular exponentiation. |
| **CLOBJ 4** | Apply number theoretic concepts in modern cryptographic systems such as RSA and Diffie–Hellman. |
| **CLOBJ 5** | Explain advanced number theory concepts such as quadratic residues and elliptic curves and their applications in cryptography and secure communication. |

f. **Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | Define the fundamental concepts of number theory including divisibility, modular arithmetic, and congruences used in computational problems. |
| **CLO 2** | Apply algorithms for prime number generation, primality testing, and integer factorization. |
| **CLO 3** | Analyze and implement efficient algorithms for modular arithmetic and modular exponentiation. |
| **CLO 4** | Apply number theoretic concepts in modern cryptographic systems such as RSA and Diffie–Hellman. |
| **CLO 5** | Explain advanced number theory concepts such as quadratic residues and elliptic curves and their applications in cryptography and secure communication. |

**Teaching & Examination Scheme:**

| **Teaching Scheme** | | | | **Evaluation Scheme** | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **L** | **T** | **P** | **C** | **Internal Evaluation** | | | **ESE** | | **Total** |
| | | | | **MSE** | **CE** | **P** | **Theory** | **P** | |
| 3 | 0 | 0 | 3 | 40 | 20 | - | 60 | - | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination **Course Content:**

| **Sr. No.** | **Topics** | **Weightage (%)** | **Teaching Hours** |
|---|---|---|---|

| | | | |
|---|---|---|---|
| **1** | Unit I: Fundamentals of Number Theory<br><br>Introduction to Computational Number Theory, Divisibility and Prime Numbers, Euclidean Algorithm and Extended Euclidean Algorithm, Greatest Common Divisor (GCD), Modular Arithmetic, Congruences, Linear Congruences and their solutions, Chinese Remainder Theorem and applications. | **20** | **8** |
| **2** | Unit II: Prime Numbers and Factorization Algorithms<br><br>Prime number generation methods, Sieve of Eratosthenes, Probabilistic and deterministic primality testing, Fermat's Little Theorem, Miller–Rabin Primality Test, Integer factorization methods: Trial division, Pollard's Rho method, Fermat factorization method. | **20** | **10** |
| **3** | Unit III: Modular Arithmetic and Exponentiation<br><br>Modular exponentiation algorithms, Fast exponentiation, Euler's Theorem, Euler's Totient Function, Carmichael function, Discrete logarithm problem, Algorithms for solving modular equations and applications in computation. | **20** | **8** |
| **4** | Unit IV: Cryptographic Applications of Number Theory<br><br>Introduction to cryptography, Public key cryptosystems, RSA algorithm, Diffie–Hellman key exchange, ElGamal cryptosystem, Digital signatures, Security issues and computational complexity of cryptographic algorithms. | **20** | **10** |

| | | | |
|---|---|---|---|
| **5** | Unit V: Advanced Topics and Applications<br><br>Quadratic residues and Legendre symbol, Jacobi symbol, Quadratic reciprocity law, Introduction to elliptic curves, Elliptic curve arithmetic, Applications of computational number theory in cryptography, coding theory and secure communication. | **20** | **6** |

**Reference Books:**


| | |
|---|---|
| 1. | Kenneth H. Rosen, Elementary Number Theory and Its Applications, Pearson. |
| 2. | **Douglas R. Stinson**, Cryptography: Theory and Practice, CRC Press. |
| 3. | Thomas Koshy, Elementary Number Theory with Applications, Academic Press. |


## Semester 7-12

a. **Course Name:** VLSI System Design

b. **Course Code:** 303105452

c. **Prerequisite:** Students should have basic knowledge of Digital Electronics, Electronic Devices and Circuits, Semiconductor Physics, and Logic Design..

d. **Rationale:** VLSI System Design is a fundamental subject for understanding the design and implementation of integrated circuits and modern electronic systems. The course introduces students to CMOS design principles, VLSI design methodologies, and system-level design approaches used in the semiconductor industry. Knowledge of VLSI design enables students to develop efficient digital integrated circuits, microprocessors, and embedded systems, which are widely used in modern computing, communication, and consumer electronics. The course prepares students for careers in chip design, semiconductor technology, and hardware system development.

**e. Course Learning Objectives:**

| | |
|---|---|
| **CLOBJ 1** | Introduce the basic concepts and design methodologies used in VLSI systems. |
| **CLOBJ 2** | Explain CMOS technology and the design of digital circuits using MOS transistors. |
| **CLOBJ 3** | Study various design styles and layout techniques used in VLSI circuits |
| **CLOBJ 4** | Analyze the performance issues such as power, delay, and area in VLSI systems. |
| **CLOBJ 5** | Define subsystem design and testing techniques used in VLSI circuits. |

**f. Course Learning Outcomes:**

| | |
|---|---|
| **CLO 1** | introduce the basic concepts and design methodologies used in VLSI systems. |
| **CLO 2** | Explain CMOS technology and the design of digital circuits using MOS transistors. |
| **CLO 3** | Study various design styles and layout techniques used in VLSI circuits. |
| **CLO 4** | Analyze the performance issues such as power, delay, and area in VLSI systems. |
| **CLO 5** | Define subsystem design and testing techniques used in VLSI circuits. |

**Teaching & Examination Scheme:**

| Teaching Scheme | | | | Evaluation Scheme | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| L | T | P | C | Internal Evaluation | | | ESE | | Total |
| | | | | MSE | CE | P | Theory | P | |
| 3 | 0 | 0 | 3 | 40 | 20 | - | 60 | - | 100 |

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation; **CE-** Continuous Evaluation; **ESE-** End Semester Examination **Course Content:**

| Sr. No. | Topics | Weightage (%) | Teaching Hours |
|---|---|---|---|
| 1 | **Unit I: Introduction to VLSI Design**<br><br>Introduction to VLSI Technology, VLSI Design Flow, Moore's Law, Design Hierarchy, VLSI Design Methodologies, IC Fabrication Process, Basic CMOS Technology, MOS Transistor Characteristics, CMOS Inverter and its operation. | **20** | **8** |
| 2 | **Unit II: CMOS Logic Design**<br><br>Static CMOS Logic, Dynamic CMOS Logic, Pass Transistor Logic, Transmission Gates, CMOS Combinational Logic Design, CMOS Sequential Logic Design, Design of Flip-Flops and Latches, Clocking strategies in VLSI circuits. | **20** | **10** |
| 3 | **Unit III: VLSI Design Styles and Layout**<br><br>Full Custom Design, Standard Cell Design, Gate Array Design, FPGA based design, Stick Diagrams, Layout Design Rules, Design Rule Checking (DRC), Layout versus Schematic (LVS), Introduction to CAD tools for VLSI design. | **20** | **8** |
| 4 | **Unit IV: VLSI Subsystem Design**<br><br>Design of Arithmetic Circuits (Adders, Multipliers), Data Path Design, Memory | **20** | **10** |

| | | | |
|---|---|---|---|
| | Elements and Memory Arrays, Programmable Logic Arrays (PLA), Programmable Logic Devices (PLD), Interconnects and Routing. | | |
| **5** | **Unit V: Testing and Performance Issues in VLSI**<br><br>Power Dissipation in CMOS Circuits, Delay and Timing Analysis, Performance Optimization Techniques, Fault Modeling, VLSI Testing Techniques, Built-In Self-Test (BIST), Design for Testability (DFT). | **20** | **6** |

**Reference Books:**


| | |
|---|---|
| 1. | Sung-Mo Kang and Yusuf Leblebici, CMOS Digital Integrated Circuits: Analysis and Design, McGraw Hill. |
| 2. | **Jan M. Rabaey**, Digital Integrated Circuits: A Design Perspective, Pearson. |
| 3. | Mead and Conway, Introduction to VLSI Systems, Addison-Wesley. |


## Semester 8-1

| | |
|---|---|
| Internship | 14 |