# SYLLABUS FOR UNDER GRADUATE PROGRAMME IN COMPUTER SCIENCE

## (Bachelor of Science Examination)

## UNDER

# CHOICE BASED CREDIT SYSTEM

**with**
**Course Rationale and Learning Based Outcomes**

ACADEMIC YEAR 2021-2022

**Department of Computer Science**
**Gangadhar Meher University**
**Sambalpur-768004**


## About the School

Bachelor of Science program in Computer Science was started by School of Physics in the year 1998. Later in the year 2001, M.Sc. in Computer Science was introduced. Computer Science got its own identity as a fully independent department in the year 2017.The School started the Ph.D. program in the year 2020 after the recruitment of three faculty members along with supportive staffs. The courses of all the programs of the school are framed under choice based credit system (CBCS) &Learning Outcomes-Based Curriculum Framework (LOCF).As per the new course curriculum the students of MSc are encouraged to take two NPTEL courses of minimum 8 weeks duration for enhancement of their academic career .In every academic session seminar, conferences and special lectures are regularly organized. The school has a seminar library containing near about 1200 books spanning all the courses. It has two computer laboratories that facilitate hands on to BSc and MSc students. The teaching in all programs emphasizes on fundamental principles, development of creative thinking and the analytical ability to solve real life problems. The School also encourages its students to engage in extra-curricular and co-curricular activities, personality development, developing team spirit, and developing organizational skills. The School has produced high-quality technocrats for the last few decades to cater to the needs of the hardware and software industry, R&D organizations, start-ups and academic institutions. Well-qualified faculties strive to impart quality teaching to establish a solid foundation for the students. Faculties are involved in diversified advanced research areas such as:

- Information Security
- Data Communication and Networking
- Machine Learning
- Soft Computing
- Pattern Recognition

- Data Analytics
- Internet of Things
- Formal Language Design and Processing

Alumni members of the School hold prestigious positions in leading software and hardware companies, academia, public service in India and abroad. The vision of the school is to produce computer science graduates with sound knowledge and skills to address current professional challenges.

# B.Sc. in Computer Science under Choice Based Credit System (CBCS) with Learning Outcomes

## About the Program

BSc Computer Science is a 3-year undergraduate program that deals with the subjects and topics related to computer science, computer application, and its services. The main aim of this program is to create quality professionals and research fellows who can work in every sector of the world by implementing the technology of computer systems and software. The courses in the program provide students a great deal to enter the real world where ambitious computer science professionals can showcase their skills as Software Developer, System Integrator, and System Analyst among other job profiles. The curriculum of the program has been designed to help students in understanding and developing computer programs in the areas related to core concepts and the emerging technologies.

The syllabus is intended to nurture academic curiosity and industry readiness in students. It has been developed to satisfy the graduate attributes and the program outcomes. Core courses include programming, data structures, algorithms, database systems, computer architecture, networking and security. Project based study help students develop competencies in a broad spectrum of topics in artificial intelligence, data analytics, web applications, and software engineering. Beyond building core competencies and technical

know-how of students, the program fosters creative skills to thrive in a dynamic world, develops leadership and social responsibility, and cultivates lifelong learning skills. The program prepares students for long term success.

**Graduate Attributes in Computer Science**

Graduate Attributes (GA) are the qualities, skills and understandings that students should develop during their time with the Higher Educational Institutions (HEI). These are qualities that also prepare graduates as agents of social good in future. Attributes of Computer Science graduate under the outcome-based teaching-learning framework may encompass the following:

- **Knowledgeable in Computer Science:** Computer Science graduates are expected to know the fundamental principles of computer science that will provide the basis for future learning and enable them to adapt to the constant rapid development of the field. These fundamental concepts would reflect the latest understanding of the field, and therefore, are dynamic in nature and require frequent and time-bound revisions.
- **Effective Communicator:** Computer Science graduates are expected to possess minimum standards of communication skills expected of a graduate in the country. They are expected to have the ability communicate effectively with a wide range of audience, comprehending and writing effective reports and design documentation, making effective oral presentations, and giving and receiving clear oral instructions.
- **Problem Solver:** Computer Science graduates are expected to identify a problem, analyze using design thinking techniques, and evolve innovative approaches for solving it.
- **Life Long Learner:** Computer Science Graduates are expected to possess basic skills of ability to self-learn and engage in life-long learning and upgrade technical skills.

- **Logical Skill**: Computer Science Graduates are expected logically think the problems at hand and analyze the domain to give a solution through logical analysis of the solution space.
- **Digital Realization**: Computer Science Graduates are expected realize and visualize the conceptual solutions of a given real life problem.
- **Software Designer:** Computer Science Graduates are expected design computer programs or software for the given solution for the problem at hand.
- **Leadership:** Computer Science Graduates are expected to demonstrate leadership abilities by implementation of thoughtful processes, driving employees to achieve the workplace goals and taking visionary initiatives.
- **Research Skill:** Computer Science Graduates are expected to identify problems, search computational resources to address the problem, validate the resources for quality and relevance and propose an effective solution.
- **Ethical:** Computer Science Graduates are expected to address ethical issues in design, use and management of information technology.

## Qualification Descriptors for a Bachelor's Degree Program in Computer Science

On completion of B.Sc. with Computer Science, the expected learning outcomes for a student are to demonstrate the following:

- **Analyzing a Problem:** Identify, analyze complex problems and formulate the computing requirements appropriate to its solution using principles of mathematics, natural sciences, and computer science.
- **Frame Solutions:** Design solutions for complex problems and develop system components that meet the specified needs with appropriate consideration of feasibility.
    - **Communication:** Communicate effectively with a range of audiences, being able to draft effective reports, and make effective visualizations.

- **Computational Knowledge**: Apply the knowledge of mathematics, science, computer science, and logical thinking to the solution of complex computational problems.
- **Modern Tool Design and Usage:** Create, select, and apply appropriate techniques, resources, and modern IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
- **The Science and Society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional scientific practice.
- **Real Life Problems**: The ability to synthesize the acquired knowledge, understanding and experience for a better and improved comprehension of the real-life problems
- **Team Work**: Ability of working in teams to build software systems.

## Programme Outcomes of B.Sc.Programme

The programme outcomes and attributes represent the knowledge, skills and attitudes the computer science graduate students of Gangadhar Meher University should have at the end of a program. The programme learning outcomes focus on various aspects of knowledge and skills that prepare students for further study, employment, and citizenship. Therefore, the Computer Science UG programme of the Gangadhar Meher University has been designed with the objective to develop in-depth knowledge of students in frontier areas of concerned subject and seeks to achieve the following:

**PO1. Disciplinary Knowledge**: Undergraduate students will demonstrate knowledge and understanding of one or more discipline knowledge.

**PO2. Critical Thinking**: Apply their analytical thoughts to evaluate evidences, claims, theories and arguments of the discipline; formulate coherent arguments, evaluate practices and policies in the academic field

**PO3. Problem Solving**: Solve the non-familiar problems based upon his/her knowledge and understanding about the discipline knowledge, including real life problems

**PO4. Communication Skills**: Ability to express thoughts and ideas effectively verbally and written. Develops the capacity to listen patiently, express himself or herself.

**PO5. Research Related Skills**: A sense of enquiry, and capacity for asking relevant/appropriate questions. Ability to recognise cause-effect relationship, define problem, formulate hypothesis, analyse, interpret and draw conclusions, generate and test hypothesis, conduct experiment and draw conclusion from findings scientifically.

**PO6. Co-Operative/Team Work**: Ability to work effectively and respectfully with diverse team, facilitate or co-ordinate efforts within group activities, work together within a team

**PO7. Multicultural Competencies**: Possess knowledge of the values and beliefs of multiple culture and a global perspective; capacity to work effectively in multiple socio-cultural context and interact respectfully with diverse social groups

**PO8. Digital Literacy**: Capacity to use ICT in a variety of learning situations, demonstrate ability to access, evaluate and use variety of relevant information sources; use appropriate software for data analysis

## Programme Specific Learning Outcome

The program enables students to achieve the following outcomes, by the time of graduation:

**PSO1**: Develop an ability to understand the theoretical foundations of computer science for designing efficient methodologies along with the knowledge of limitations of computing..

**PSO2**: Identify existing models and analyse complex computer science problems reaching substantial conclusions using the formulation of algorithms, techniques and data structures.

**PSO3**: Apply principles and practices of Computer Science involving mathematical modelling for finding solutions to real time challenges.

| Course Code | Course Title | | Programme Outcomes (POs) | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | PSO 1 | PSO 2 | PSO 3 | |
| CC-1 | Programming Using C | ✔ | ✔ | ✔ | | ✔ | | | ✔ | ✔ | ✔ | ✔ | |
| CC-2 | Digital Logic | ✔ | ✔ | ✔ | | ✔ | | | ✔ | ✔ | ✔ | ✔ | |
| GE-1 | Computer Fundamentals | ✔ | ✔ | ✔ | | | | | ✔ | ✔ | ✔ | ✔ | |
| CC-3 | Programming Using C++ | ✔ | ✔ | ✔ | | | | | ✔ | ✔ | ✔ | ✔ | |
| CC-4 | Data Structures | ✔ | ✔ | ✔ | | | | | ✔ | ✔ | ✔ | ✔ | |
| GE-2 | C and Data structures | ✔ | ✔ | ✔ | | | | | ✔ | ✔ | ✔ | ✔ | |
| CC-5 | JAVA Programming | ✔ | ✔ | ✔ | | ✔ | | | ✔ | ✔ | ✔ | ✔ | |
| CC-6 | Database Systems | ✔ | ✔ | ✔ | | ✔ | | | ✔ | ✔ | ✔ | ✔ | |
| CC-7 | Discrete Mathematical Structures | ✔ | ✔ | ✔ | | ✔ | | | ✔ | ✔ | ✔ | ✔ | |
| GE-3 | Programming in Python | ✔ | ✔ | ✔ | | ✔ | | | | ✔ | ✔ | ✔ | |
| CC-8 | Operating Systems | ✔ | ✔ | ✔ | | ✔ | | | ✔ | ✔ | ✔ | ✔ | |
| CC-9 | Computer Networks | | ✔ | ✔ | | ✔ | | | ✔ | ✔ | ✔ | ✔ | |
| CC-10 | Computer Graphics | ✔ | ✔ | ✔ | | | | | | ✔ | ✔ | ✔ | |
| GE-4 | Web Technology | ✔ | ✔ | ✔ | | | | | ✔ | ✔ | ✔ | ✔ | |
| CC-11 | Web Technology | ✔ | ✔ | ✔ | | | | | ✔ | ✔ | ✔ | ✔ | |
| CC-12 | Software Engineering | | ✔ | ✔ | | ✔ | | | ✔ | ✔ | ✔ | ✔ | |
| DSE-1 | DSE-1 | ✔ | ✔ | ✔ | | | | | | ✔ | ✔ | ✔ | |
| DSE-2 | DSE-2 | ✔ | ✔ | ✔ | | ✔ | | | ✔ | ✔ | ✔ | ✔ | |
| CC-13 | Artificial Intelligence | | ✔ | ✔ | | ✔ | | | ✔ | ✔ | ✔ | ✔ | |
| CC-14 | Algorithm Design Techniques | ✔ | ✔ | ✔ | | ✔ | | | ✔ | ✔ | ✔ | ✔ | |
| DSE-3 | DSE-3 | ✔ | ✔ | ✔ | | ✔ | | | | ✔ | ✔ | ✔ | |
| DSE-4 | Project work / Dissertation | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ | | ✔ | ✔ | ✔ | ✔ | |
| | Data Mining | ✔ | ✔ | ✔ | | | | | | ✔ | | ✔ | |
| VA-1 | Certificate Course in Machine Learning Using Python | ✔ | ✔ | ✔ | | | | | | ✔ | ✔ | ✔ | |

## Mapping Course with Programme Specific Outcome (PSO) Computer Science

### Teaching Learning Process

The teaching-learning process of the programme focuses on learner-centric pedagogies and interactive and participatory pedagogies. Planning for teaching constitutes a critical dimension. This process is a well-structured and sequenced acquisition of knowledge and skills. This process involves classroom lectures, tutorials, and laboratory experiments. The tutorials allow closer interaction between the students and the teacher as each student gets individual attention. For visual and better learning outcomes instructors would use ICT facilities, e-learning platforms, and other innovative e-content platforms for student-centric learning methods. As part of participative teaching-learning practices, few courses include seminars and presentations. To cater to the needs of slow and advanced learners, special, remedial, and peer teaching classes are implemented. Apart from these, special lectures by invited experts, workshops, and National/International seminars are arranged to augment knowledge, encourage innovative ideas and expose the students to global academic and research advancement. There is also the provision of a mentor-mentee for students where the mentor can interact with students periodically to understand the progress and challenges of the student so that preventive action or suggestions can be taken to resolve the issues. For the realization of real-life application of the knowledge acquired, project works are offered in the final semester, which is an integral component of the programme. To develop teamwork spirit among students, most of the project works are assigned in a group. To develop technical writing and presentation skills, the students will be asked to submit written assignments and projects report followed by oral presentations, which are made as part of the course. For each course, students will be assigned regular home assignments and will be tested periodically through quizzes and class tests to ensure that they have properly learned the course material.

## Assessment Methods/Evaluation Scheme

Assessment methods and evaluation schemes will be as per the Gangadhar Meher University Examination rules and regulation. Each Core, Discipline Specific Electives (DSE) and Generic Electives (GE) papers will have maximum 100 marks with 6 credits whereas Ability Enhancement (AECC) and Skill Enhancement Courses (SEC) will have maximum 50 marks with 4 credits. **The value added and add on courses if any will have maximum 50 marks with 2 credits**.

The distribution of marks for assessment and evaluation is follows:

**For 100 Marks Paper:**

I. 15 marks will be assessed and evaluated by internal assessment.

II. 25 Marks will be assessed and evaluated by practical/laboratory experiments.

III. 60 marks will be assessed and evaluated by end term examination

**For 50 Marks Paper:**

I. 10 marks will be assessed and evaluated by internal assessment

II. 40 marks will be assessed and evaluated by end term examination.

**For 25 marks Paper:**

Total marks will be assessed and evaluated by end term examination.

***The internal assessment*** and evaluation may be done through a continuous process having assignments, class test, review works, writing of term papers, quiz or examination or lab experiments.

***Practical Test*** will be conducted through submission of practical records/laboratory experiments/specimen identification and evaluated by internal/external examiners.

***The end term examination*** will be based on written examination for theory paper and the question will be set following Bloom's taxonomy which will cover the entire syllabus of the course.

***For the research project paper***, the student will submit the project write up and defend his/her project through presentation/Viva-Voce as per the University rule.

| COURSE STRUCTURE OF UG COMPUTER SCIENCE UNDER CBCS | | | | |
|---|---|---|---|---|
| **Semester** | **Course Code** | **Course Name** | **Credits** | **Total Marks** |
| I | AECC-1 | Environmental Studies | 4 | 100 |
| | EV-1(Ethics and Values) | Issues related to Women | 1 | 25 |
| | CC-1(Theory)<br>CC-1(Practical) | Programming Using C<br>Programming Using C LAB | 4<br>2 | 75<br>25 |
| | CC-2 (Theory)<br>CC-2 (Practical) | Digital Logic<br>Digital Logic LAB | 4<br>2 | 75<br>25 |
| | GE-1 (Theory)<br>GE-1 (Practical) | Computer Fundamentals<br>Computer Fundamentals LAB | 4<br>2 | 75<br>25 |
| | | **Total** | **23** | **450** |
| II | AECC-II | MIL(Odia/Hindi/English) | 4 | 100 |
| | EV-2(Ethics and Values) | And Good Citizenship | 1 | 25 |
| | CC-3(Theory)<br>CC-3(Practical) | Programming Using C++<br>Programming Using C++ LAB | 4<br>2 | 75<br>25 |
| | CC-4 (Theory)<br>CC-4 (Practical) | Data Structures<br>Data Structures LAB | 4<br>2 | 75<br>25 |
| | GE-2 (Theory)<br>GE-2 (Practical) | C and Data Structures<br>C and Data Structures Lab | 4<br>2 | 75<br>25 |
| | | **Total** | **23** | **450** |
| III | CC-5(Theory)<br>CC-5(Practical) | JAVA Programming<br>JAVA Programming LAB | 4<br>2 | 75<br>25 |
| | CC-6(Theory)<br>CC-6(Practical) | Database Systems<br>Database Systems LAB | 4<br>2 | 75<br>25 |
| | CC-7(Theory)<br>CC-7(Practical) | Discrete Mathematical Structures<br>Discrete Mathematical Structures | 4<br>2 | 75<br>25 |
| | SEC-1 | **Student are supposed to read the subject as given by University** | 4 | 100 |
| | EV-3(Ethics and Values) | Issues of Drug, Tobacco and Alcohol Addiction | 1 | 25 |
| | GE-3 (Theory)<br>GE-3 (Practical) | Programming in Python<br>Programming in Python LAB | 4<br>2 | 75<br>25 |
| | | **Total** | **29** | **525** |
| | CC-8(Theory)<br>CC-8(Practical) | Operating Systems<br>Operating Systems LAB | 4<br>2 | 75<br>25 |
| | CC-9(Theory)<br>CC-9(Practical) | Computer Networks<br>Computer Networks LAB | 4<br>2 | 75<br>25 |
| | CC-10(Theory)<br>CC-10(Practical) | Computer Graphics<br>Computer Graphics LAB | 4<br>2 | 75<br>25 |

| | | | | |
|---|---|---|---|---|
| IV | SEC-II* | **Students will choose a paper from a group of papers offered by University/Department.** | 4 | 100 |
| | GE-4 (Theory)<br>GE-4 (Practical) | Web Technology<br>Web Technology LAB | 4<br>2 | 75<br>25 |
| | EV-4(Ethics and Values) | Ethical Values for Students life | 1 | 25 |
| | | **Total** | **29** | **525** |
| V | CC-11(Theory)<br>CC-11(Practical) | Web Technology<br>Web Technology LAB | 4<br>2 | 75<br>25 |
| | CC-12(Theory)<br>CC-12(Practical) | Software Engineering<br>Software Engineering LAB | 4<br>2 | 75<br>25 |
| | DSE-1 (Theory)<br>DSE-1 (Practical) | Numerical Techniques<br>Numerical Techniques LAB | 4<br>2 | 75<br>25 |
| | DSE-2 (Theory)<br>DSE-2 (Practical) | Unix Shell Programming<br>Unix Shell Programming LAB | 4<br>2 | 75<br>25 |
| | EV-5(Ethics and Values) | Vulnerable section of society: Understanding their issues | 1 | 25 |
| | | **Total** | **25** | **425** |
| VI | CC-13(Theory)<br>CC-13(Practical) | Artificial Intelligence<br>Artificial Intelligence LAB | 4<br>2 | 75<br>25 |
| | CC-14(Theory)<br>CC-14(Practical) | Algorithm Design Techniques<br>Algorithm Design Techniques LAB | 4<br>2 | 75<br>25 |
| | DSE-3 (Theory)<br>DSE-3 (Practical) | Data Science<br>Data Science LAB | 4<br>2 | 75<br>25 |
| | DSE-4 | Project Work/ Dissertation | 6 | 100 |
| | DSE-4 (Theory)<br>DSE-4 (Practical) | Data Mining<br>Data Mining LAB | 4<br>2 | 75<br>25 |
| | EV-6(Ethics and Values) | Environmental and Techno Ethics | 1 | 25 |
| | | **Total** | **25** | **425** |
| | | **Grand Total** | **154** | **2750** |

| **Certificate Course Offered beyond the Prescribed Syllabus** | | | | | |
|---|---|---|---|---|---|
| **Semester** | **Course Code** | **Course Name** | **Credits** | **Total Marks** | **Contact Hours** |
| Any Semester | | Certificate Course in Machine Learning using Python | | | |

## Semester-I

# CC-1: Programming Using C

**Credits: 4**
**Full Mark 75 (End Term 60 + Internal 15)**

### COURSE OVERVIEW:

This course introduces fundamentals of computer programming using the C programming language to the students. Starting with simple programs, the course will cover advanced topics like structures, pointers, file processing and pre-processor directives etc. and enable the students to write programs using C language for solving various engineering problems.

### COURSE OBJECTIVES:

1. To learn basics of programming using c language.
2. To be able to write programs/applications in C.
3. Develop logical thinking to solve computational problems.

| **Core Course-1** | |
|---|---|
| **Programming Using C** | |
| **UNIT-I:** | **10hrs** |
| Introduction: Introduction to Programming Language, Introduction to C Programming, Keywords & Identifiers, Constants, Variables, Input and Output Operations, Compilation and pre-processing,Datatypes:Differentdatatypes,Datatypesqualifier,modifiers,Memory representation, size and range, Operators: Operators (Arithmetic, Relational, Logical, Bitwise, Assignment & compound assignment, Increment & Decrement, Conditional),Operator types (unary, binary, ternary). Expressions, Order of expression (Precedence and associativity) Control structures: Decision Making and Branching (Simple IF Statement, IF…ELSE Statement, Nesting IF… ELSE Statement, ELSE IF Ladder), Selection control structure (Switch Statement). | |
| **UNIT-II:** | **10hrs** |
| Loops: The WHILE Statement, The DO…WHILE Statement, The FOR Statement, Jumps in Loops, Array: Concept of Array, Array Declaration, types of array (one and multiple dimension), Character Arrays and Strings, Subscript and pointer representation of array, Array of Pointers, Limitation of array, Pointers: Concept of Pointer(null pointer, wild pointer, dangling pointer, generic pointer),Pointer Expressions, Accessing the Address of a Variable, Declaring Pointer Variables, Initializations of Pointer Variable, Accessing a Variable through its Pointer, Pointer arithmetic. | |
| **UNIT-III:** | **10hrs** |
| Storage class: Types (auto, register, static, extern), scope rules, declaration and definition. Function: Function & types(User defined function, library function)Function Definition, Declaration, Function Calls, Header file and library, Function Arguments, string handling function (strlen, strcmp, strcpy, strncpy, strcat, strstr), Function recursion, Functions Returning Pointers, Pointers to Functions, Command line arguments, Application of pointer(dynamic memory allocation). | |

| **UNIT-IV:** | **10hrs** |
|---|---|
| Structure and Union: Defining, Declaring, Accessing, Initialization Structure, nested structure, self-referential structure, bit-field, Arrays of Structures, Structures and Functions, Unions, difference between structure and union, active data member, structure within union, Self-referential Structure, File: File Management in C, Defining and Opening a File, File opening modes (read, write, append), Closing a File, File operations, file and stream, Error Handling During/O Operations, sequential and random access file, low level and high level file. | |
| **Text Book:** | |
| *1.* E. Balagurusamy, "Programming in ANSI C", McGraw-Hill Education, 7th Edition, 2017.<br>2. Y. Kanetker, Let Us C, 16th Edition, BPB Publications, 2018. | |
| **Reference Book:** | |
| 1. B. W. Kernighan and D. M. Ritchie, The C Programming Language, 2nd Edition, Pearson Education, 2015.<br>2. H. Schildt, C: The Complete Reference, 4th Edition, McGraw-Hill, 2017.<br>3. A. Kelley and I. Pohl, A Book on C, 4th Edition, Pearson Education, 2008.<br>4. B. Gottfried, Schaum's Outline of Programming with C, 3rd Edition, McGraw-Hill, 2017. | |

| | **COURSE OUTCOMES:**<br>After completion of this course successfully, the students will be able to- |
|---|---|
| **CO1** | Formulate logic of a problem and write C programs using variables, expressions and input/output statements. |
| **CO2** | Develop structured C programs involving decision making using different control constructs |
| **CO3** | Solve problems involving similar set of data items and convert them into C programs using arrays. Design modular C programs and handle heterogeneous data items using structures & unions. |
| **CO4** | Write C applications using pointers, pre-processor directives, command line arguments and files. |

**CORE COURSE–1 Practical: Programming Fundamentals using C Lab (Credits: 2) Marks:25**

1. Write a Program to find greatest among three numbers.
2. Write a Program to all arithmetic operation using switch case.
3. Write a Program to print the sum and product of digits of an integer.
4. Write a Program to reverse a number.
5. Write a Program to compute the sum of the first in terms of the following series S = 1+1/2+1/3+1/4+……
6. Write a Program to compute the sum of the first in terms of the following series S=1-2+3-4+5…….
7. Write a function that checks whether a given string is Palindrome or not. Use this function to find whether the string entered by user is Palindrome or not.
8. Write a function to find whether a given no. is prime or not. Use the same to generate the prime numbers less than 100.
9. Write a Program to compute the factors of a given number.

10. Write a program to swap two numbers using macro.
11. Write a Program to print a triangle of stars as follows(take number of lines from user):

    *

    ***

    *****

    *******
12. Write a Program to perform following actions on an array entered by the user:
    a) Print the even-valued elements
    b) Print the odd-valued elements
    c) Calculate and print the sum and average of the elements of array
    d) Print the maximum and minimum element of array
    e) Remove the duplicates from the array
    f) Print the array in reverse order

    The program should present a menu to the user and ask for one of the options. The menus should also include options to re-enter array and to quit the program.
13. Write a Program that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.
14. Write a program that swaps two numbers using pointers.
15. Write a program in which a function is passed address of two variables and then alter its contents.
16. Write a program which takes the radius of a circle as input from the user, passes it to another function that computes the area and the circumference of the circle and displays the value of area and circumference from the main( ) function.
17. Write a program to find sum and average of n elements entered by the user. To write this program, allocate memory dynamically using malloc ( ) / calloc ( ) functions.
18. Write a menu driven program to perform following operations on strings:
    a. Show address of each character in string
    b. Concatenate two strings without using strcat function.
    c. Concatenate two strings using strcat function.
    d. Compare two strings
    e. Calculate length of the string (use pointers)
    f. Convert all lower case characters to uppercase
    g. Convert all upper case characters to lowercase
    h. Calculate number of vowels
    i. Reverse the string
    j. Given two ordered arrays of integers, write a program to merge the two-arrays to get an ordered array.
19. Write a program to copy the content of one file to other.

# CC-2: DIGITAL LOGIC

**Credits: 4**
**Full Mark 75 (End Term 60+Internal-15)**

## COURSE OVERVIEW:

The course addresses the concepts, principles and techniques of designing digital systems. It teaches the fundamentals of digital systems applying the logic design and development techniques. This course forms the basis for the study of advanced subjects like Computer Architecture and Organization, Microprocessor through Interfacing, VLSI Designing.

## COURSE OBJECTIVES:

1. To understand different methods used for the simplification of Boolean functions and Binary arithmetic.
2. To design and implement combinational circuits, synchronous & asynchronous sequential circuits.
3. To study in detail about Semiconductor Memory Systems.

| **Core Course-2** | |
|---|---|
| **Digital Logic** | |
| **UNIT-I:** | **10hrs** |
| Character Codes, Decimal System, Binary System, Decimal to Binary Conversion, Hexadecimal Notation, Boolean Algebra, Basic Logic Functions: Electronic Logic Gates, Synthesis of Logic Functions, Minimization of Logic Expressions, Minimization using Karnaugh Maps, Synthesis with NAND and NOR Gates, Tri-State Buffers | |
| **UNIT-II:** | **10hrs** |
| Arithmetic: Addition and Subtraction of Signed Numbers, Addition/ Subtraction Logic Unit, Design of Fast Adders: Carry-Look ahead Addition, Multiplication of Positive Numbers, Signed-Operand Multiplication: Booth Algorithm, Fast Multiplication: Bit-Pair Recoding Multipliers, Carry-Save Addition of Summands, Integer Division, Floating-Point Numbers and Operations: IEEE Standard for Floating-Point Numbers, Arithmetic Operations on Floating-Point Numbers, Guard Bits and Truncation, Implementing Floating-Point Operations. | |
| **UNIT-III:** | **10hrs** |
| Flip-Flops, Gated Latches, Master-Slave Flip-Flops, Edge-Triggering, T Flip-Flops, JK Flip-Flops. Registers and Shift Registers, Counters, Decoders, Multiplexers, Programmable Logic Devices (PLDs), Programmable Array Logic (PAL), Complex Programmable Logic Devices (CPLDs), FieldProgrammable Gate Array (FPGA), Sequential Circuits, UP/ DOWN Counters, Timing Diagrams, The Finite State Machine Model, Synthesis of Finite State Machines. | |
| **UNIT-IV:** | **10hrs** |
| Memory System: Semiconductor RAM Memories, Internal Organization of Memory Chips, Static Memories, Asynchronous DRAMS, Synchronous DRAMS, Structure of Large Memories, Memory, System Considerations, RAMBUS Memory. Read-Only Memories: ROM, PROM, EPROM, EEPROM, Flash Memory, Speed, Size, and Cost of Memory. Secondary Storage: Magnetic Hard Disks, Optical | |

| |
|---|
| Disks, Magnetic Tape Systems. |
| **Text Book:** |
| 1. Carl Hamacher, Zvonko Vranesic and Safwat Zaky, *"*Computer Organization*"*, Tata McGraw Hill, Fifth Edition, 2002 |
| **Reference Book:** |
| 2. M.Morris Mano,"Digital logic and Computer Design", Pearson Education India, First Edition ,2017. |

| | **COURSE OUTCOMES:**<br>After completion of this course successfully, the students will be able to- |
|---|---|
| **CO1** | Understand various types of number systems and their conversions. |
| **CO2** | Identify the importance of canonical forms in the minimization or other optimization of Boolean formulas in general and digital circuits. |
| **CO3** | Minimize functions using any type of minimizing algorithms (Boolean algebra, Karnaugh map or Tabulation method). |
| **CO4** | Analyze the design procedures of Combinational and Sequential circuits |

**CORE COURSE–2 Practical: Digital Logic Lab (Credits: 2)**
**Marks:25**

1. **Introduction to Xilinx Software(VHDL)**
2. **Write the VHDL code for**
   a. Realizing all logic gates.
   b. Combination Circuit.
   c. ADDER.
   d. SUBTRACTOR.
   e. MUX.
   f. DE-MUX.
   g. Encoder.
   h. Decoder.
   i. PAL.
   j. PLA.
3. **Write the VHDL program for the following Sequential Logic Circuits**
   a. Flip Flops. b. Shift Registers. c. Counters. d. Memory Elements.

## Semester-II

# CC-3: PROGRAMMING USING C++

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

### COURSE OVERVIEW:

The course lets the learners know about object oriented concepts. It contains basic programming techniques using C++. Implementation of object oriented concepts using C++ language is elaborated in the course. It serves a fundamental guide to solve real world problems using object oriented programming structures.

### COURSE OBJECTIVES:

1. To know about the Object Oriented Programming concepts.
2. To learn basics of C++ programming language.
3. To be able to develop skills to design applications in C++ for solving computational problems.

| **Core Course-3** |
|---|
| **PROGRAMMING USING C++** |
| **UNIT-I:** **10hrs** |
| Principles of Object-Oriented Programming: Object-Oriented Programming (OOP) Paradigm, Basic Concepts of OOP, Benefits of OOP, Characteristics of OOPS, Object Oriented Languages, Applications of OOP.<br>Introduction to C++, Difference between C & C++, Tokens, Data types, Operators, Structure of C++ Program, C++ statements, Expressions and Control Structures.<br>Functions in C++: Argument passing in function, Inline Functions, Default Arguments, Const. Arguments, Friend function. |
| **UNIT-II:** **10hrs** |
| Classes and Objects: Defining Member Functions, Making an outside Function Inline, Nested Member Functions, Private Member Functions, Arrays within a Class, Memory Allocation for Objects, Static Data Members, Static Member Functions, Arrays of Objects, Objects as Function Arguments, Friend Functions.<br>Constructors & Destructors: Constructors, Parameterized Constructors, Constructors with Default Arguments, Dynamic Initialization of Objects, Copy Constructor, Dynamic Constructors, Destructors. |
| **UNIT-III:** **10hrs** |
| Inheritance: Basics of Inheritance, Type of Inheritance, Virtual Base Classes, Abstract Classes, Member Classes, Nesting of Classes. Polymorphism: Pointers, Pointers to Objects, this Pointer, Pointers to Derived Classes, Virtual Functions, Pure Virtual Functions, Function Overloading, Operator Overloading. |
| **UNIT-IV:** **10hrs** |
| Managing Console I/O Operations: C++ Streams, C++ Stream Classes, Unformatted I/O Operations, Formatted Console I/O Operations, Managing Output with Manipulators. Files: Classes for File Stream Operations, Opening and Closing a File, Detecting end-of-file, |

| |
|---|
| File Modes, File Pointers and their Manipulations, Sequential Input and Output Operations, Updating a File: Random Access, Error Handling during File Operations, Command-line Arguments. |
| **Text Books:** |
| 1. E. Balgurusawamy, “Object Oriented Programming with C++”, TMH, Eighth Edition, 2020.<br>2. Paul Deitel, Harvey Deitel, "C++: How to Program", Prentice Hall, Tenth Edition, 2020. |
| **Reference Books:** |
| 1.Bjarne Stroustroup,” Programming - Principles and Practice using C++”, Addison-Wesley , Second Edition, 2014<br>2.Herbtz Schildt, “C++: The Complete reference”, MGH, Fourth Edition, 2017.<br>3.P. C. Sethi, P. K. Behera, “Programming in C++”, Kalyani Publisher, 2018. |

| | **COURSE OUTCOMES:**<br>After completion of this course successfully, the students will be able to- |
|---|---|
| **CO1** | Understand the differences between function oriented programming and object oriented programming. |
| **CO2** | Learn the C++ programming structures to code an algorithm. |
| **CO3** | Identify appropriate programming units suitable to solve a complex problem. |
| **CO4** | Develop software applications for solving real world problems. |

**CORE–3 Practical: Programming using C++ Lab (Credits: 2)**
**Marks: 25**

1. Write a Program to find greatest among three numbers using nested if…else statement.
2. Write a Program to check a number is prime or not.
3. Write a Program to find the GCD and LCM of two numbers.
4. Write a program to print the result for following series: 1! + 2! + 3! + ……
5. Write a program to print multiplication table from 1 to 10.
6. Write a Program for Swapping of two numbers using pass by value.
7. Write a Program for Swapping of two numbers using pass by address.
8. Write a Program for Swapping of two numbers using pass by reference.
9. Write a Program to find sum of four numbers using default argument passing.
10. Write a Program to find square and cube of a number using inline function.
11. Write a Program to find the factorial of a number.
12. Write a Program to find reverse of a number.
13. Write a program to find sum of four numbers using default argument passing in member function.
14. Write a Program to find area of circle, triangle and rectangle using function overloading.
15. Write a program to distinguish the properties of static and non-static ata members.
16. Write a program to show the method of accessing static private member function.
17. Write a program to show the ways of calling constructors and destructors.
18. Write a program to perform ++ operator overloading using member function.
19. Write a program to perform ++ operator overloading using friend function.
12
20. Write a program to perform + operator overloading for two complex number addition.
21. Write a program to perform + operator overloading for string concatenation.
22. Write a program to perform single inheritance.
23. Write a program to perform multiple inheritances.

24. Write a program to create an integer array using new operator and find the sum and average of array elements.
25. Write a program to implement virtual destructor.
26. Create the Person class. Create some objects of this class (by taking information from the user). Inherit the class Person to create two classes Teacher and Student class. Maintain the respective information in the classes and create, display and delete objects of these two classes (Use Runtime Polymorphism).
27. Write a program to Copy the contents of one file to other**.**

## CC-4: DATA STRUCTURE

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

### COURSE OVERVIEW:

Data structure is important in almost every aspect where data is involved. In general, algorithms that involve efficient data structure is applied in the following areas: numerical analysis, operating system, A.I., compiler design, database management, graphics, and statistical analysis. Data structures are used in any program or software. Employability areas are Compiler Design, Operating System, DBMS, Graphics, Simulation, and Numerical Analysis.

### COURSE OBJECTIVES:

1. To learn how the choice of data structures impacts the performance of programs.
2. To study specific data structures such as arrays linear lists, stacks, queues, hash tables, binary trees, binary search trees, heaps and AVL trees.
3. To learn efficient searching and sorting techniques.

| **Core Course-4** | |
|---|---|
| **Data Structure** | |
| **UNIT-I:** | **10hrs** |
| Introduction: Basic Terminology, Data structure, Time and space complexity, Review of Array, Structures, Pointers. Linked Lists: Dynamic memory allocation, representation, Linked list insertion and deletion, Searching, Traversing in a list, doubly linked list, sparse matrices. | |
| **UNIT-II:** | **10hrs** |
| Stack: Definition, Representation, Stack operations, Applications (Infix–Prefix–Postfix Conversion & Evaluation, Recursion). Queues: Definition, Representation, Types of queue, Queue operations, Applications. | |
| **UNIT-III:** | **10hrs** |
| Trees: Tree Terminologies, General Tree, Binary Tree, Representations, Traversing, BST, Operations on BST, Heap tree, AVL Search Trees, M-way search tree, Applications of all trees. | |
| **UNIT-IV:** | **10hrs** |
| Sorting: Exchange sorts, Selection Sort, Bubble sort, Insertion Sorts, Merge Sort, Quick Sort, Radix Sort, Heap sort. | |

| |
|---|
| Searching: Linear search, Binary search. |
| **Text Book:** |
| 1. D.Samanta , “Classic Data Structure ”, Second Edition, 2009 |
| **Reference Books:** |
| 1. Ellis Horowitz, Sartaj Sahni, Susan Andrson-Freed, ”Fundamentals of Data Structures in C”, GalgotiaPublications, 2nd Edition, 2008.<br>2. SastryC.V.,Nayak R, Ch.Rajaramesh, “Data Structure & Algorithms”, I.K. International Publishing House Pvt. Ltd, New Delhi, First Edition, 2019. |

| | **COURSE OUTCOMES:**<br>After completion of this course successfully, the students will be able to- |
|---|---|
| **CO1** | Understand various types of number systems and their conversions. |
| **CO2** | Identify the importance of canonical forms in the minimization or other optimization of Boolean formulas in general and digital circuits. |
| **CO3** | Minimize functions using any type of minimizing algorithms (Boolean algebra, Karnaugh map or Tabulation method). |
| **CO4** | Analyse the design procedures of Combinational and Sequential circuits |

**CORE COURSE–4 Practical: Data Structure Lab (Credit: 2)**
**Marks:25**

1. To insert and delete elements from appropriate position in an array.
2. To search an element and print the total time of occurrence in the array.
3. To delete all occurrence of an element in an array.
4. Array implementation of Stack.
5. Array implementation of Linear Queue.
6. Array implementation of Circular Queue.
7. To implement linear linked list and perform different operation such as node insert and delete, search of an item, reverse the list.
8. To implement circular linked list and perform different operation such as node insert and delete.
9. To implement double linked list and perform different operation such as node insert and delete.
10. Linked list implementation of Stack.
11. Linked list implementation of Queue.
12. Polynomial representation using linked list.
13. To implement a Binary Search Tree.
14. To represent a Sparse Matrix.
15. To perform binary search operation.
16. To perform Bubble sort.
17. To perform Selection sort.
18. To perform Insertion sort.
19. To perform Quick sort.
20. To perform Merge sort.

## Semester-III

# CC-5: JAVA PROGRAMMING

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

### COURSE OVERVIEW:

The course builds on the skills gained by students in Java Fundamentals or Java Foundations to help advance Java programming skills. Students will design object-oriented applications with Java and will create Java programs using hands-on, engaging activities

### COURSE OBJECTIVES:

1. To learn the fundamentals of Object Oriented Programming in Java environment.
2. To learn the use of Java language and the Java Virtual Machine.
3. To write simple Java programming applications

## Core Course-5

### Java Programming

**UNIT-I:** **10hrs**

Introduction to Java: Java History, Architecture and Features, Understanding the semantic and syntax differences between C++ and Java, Compiling and Executing a Java Program, Variables, Constants, Keywords (super, this, final, abstract, static, extends, implements, interface) , Data Types, Wrapper class, Operators (Arithmetic, Logical and Bitwise) and Expressions, Comments, Doing Basic Program Output, Decision Making Constructs (conditional statements and loops) and Nesting, Java Methods (Defining, Scope, Passing and Returning Arguments, Type Conversion and Type and Checking, Built-in Java Class Methods). Input through keyboard using Command line Argument, the Scanner class, Buffered Reader class.

**UNIT-II:** **10hrs**

Object-Oriented Programming Overview: Principles of Object-Oriented Programming, Defining & Using Classes, Class Variables & Methods, Objects, Object reference, Objects as parameters, final classes, Garbage Collection. Constructor- types of constructor, this keyword, super keyword. Method overloads and Constructor overloading. Aggregation vs Inheritance, Inheritance: extends vs implements, types of Inheritance, Interface, Up-Casting, Down-Casting, Auto-Boxing, Enumerations, Polymorphism, Method Overriding and restrictions. Package: Pre-defined packages and Custom packages.

**UNIT-III:** **10hrs**

Arrays: Creating &Using Arrays (1D,2D,3D and Jagged Array),Array of Object, Referencing Arrays Dynamically. Strings and I/O: Java Strings: The Java String class, Creating & Using String Objects, Manipulating Strings, String Immutability& Equality, and Passing Strings to & from Methods, String Buffer Classes and String Builder Classes. IO package: Understanding Streams File class and its methods, Creating, Reading, Writing using classes: Byte and Character streams, FileOutputStream, FileInputStream, FileWriter, FileReader, InputStreamReader, PrintStream, PrintWriter. Compressing and Uncompressing File.

| UNIT-IV: | 10hrs |
|---|---|
| Exception Handling, Threading, Networking and Database Connectivity: Exception types, uncaught exceptions, throw, built-inexceptions, Creating your own exceptions; Multi-threading: The Thread class and Runnable interface, creating single and multiple threads, Thread prioritization,synchronizationandcommunication,suspending/resumingthreads.Usingjava.net package, Overview of TCP/IP and Datagram programming. Accessing and manipulating databases using JDBC | |
| **Text Book:** | |
| **1.**E.Balagurusamy,"Programming with Java",Tata McGraw-Hill,Sixth Edition,2019. | |
| **Reference Book:** | |
| 1.Herbert Schildt,"Java:The Complete Reference ", Tata McGraw-Hill, Twelfth Edition,2022 | |

| | COURSE OUTCOMES:<br>After completion of this course successfully, the students will be able to- |
|---|---|
| CO1 | Use the syntax and semantics of java programming language and basic concepts of OOP. |
| CO2 | Develop reusable programs using the concepts of inheritance, polymorphism, interfaces and packages. |
| CO3 | Apply the concepts of Multithreading and Exception handling to develop efficient and error free codes |
| CO4 | Able to apply object oriented programming features and concepts for solving problem of real word scenarios. |

**CORE COURSE–5 Practical: Java Programming Lab (Credits: 2)**
**Marks:25**

1. To find the sum of any number of integers entered as command line arguments.
2. To find the factorial of a given number.
3. To convert a decimal to binary number.
4. To check if a number is prime or not, by taking the number as input from the keyboard.
5. To find the sum of any number of integers interactively, i.e., entering every number from the keyboard, whereas the total number of integers is given as a command line argument
6. Write a program that show working of different functions of String and StringBufferclasss like setCharAt( ), setLength( ), append( ), insert( ), concat( )and equals( ).
7. Write a program to create a – ―distance‖ class with methods where distance is computed in terms of feet and inches, how to create objects of a class and to see the use of this pointer .
8. Modify the–―distance‖ class by creating constructor for assigning values (feet and inches) to the distance object. Create another object and assign second object as reference variable to another object reference variable. Further create a third object which is a clone of the first object.
9. Write a program to show that during function overloading, if no matching argument is found, then Java will apply automatic type conversions (from lower to higher data type) .
10. Write a program to show the difference between public and private access specifiers. The program should also show that primitive data types are passed by value and objects are passed by reference and to learn use of final keyword.

11. Write a program to show the use of static functions and to pass variable length arguments in a function. 12. Write a program to demonstrate the concept of boxing and unboxing.
13. Create a multi-file program where in one file a string message is taken as input from the user and the function to display the message on the screen is given in another file (make use of Scanner package in this program).
14. Write a program to create a multilevel package and also creates a reusable class to generate Fibonacci series, where the function to generate Fibonacci series is given in a different file belonging to the same package.
15. Write a program that creates illustrates different levels of protection in classes/subclasses belonging to same package or different packages
16. Write a program – ―Divide By Zero that takes two numbers a and b as input, computes a/b, and invokes Arithmetic Exception to generate a message when the denominator is zero.
17. Write a program to show the use of nested try statements that emphasizes the sequence of checking for catch handler statements.
18. Write a program to create your own exception types to handle situation specific to your application (Hint: Define a subclass of Exception which itself is a subclass of Throwable).
19. Write a program to demonstrate priorities among multiple threads.
20. Write a program to demonstrate different mouse handling vents like mouseClicked(), mouseEntered( ), mouseExited( ), mousePressed( ), mouseReleased( ) &mouseDragged( ).
21. Write a program to demonstrate different keyboard handling events.

## CC-6: DATABASE SYSTEMS

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

### COURSE OVERVIEW:

This course provides an overview of architecture and applications of database systems. It includes brief information about different types of databases. The course lays a foundation of physical modelling and logical modelling in designing a database. It focuses on different aspects of database like security, authentication, normalization, and concurrency etc.

### COURSE OBJECTIVES:

1. To learn the fundamental elements of database system.
2. To understand the basic concepts of relational database management systems.
3. To know various SQL commands.
4. To analyse the tables from the context of normalization.

| Core Course-6 | |
|---|---|
| **Database Systems** | |
| **UNIT-I:** | **10hrs** |
| Introduction to Database and Database Users, Database System Concepts and Architecture: data | |

| |
|---|
| Models, schema, and instances, Conceptual Modeling and Database Design: Entity Relationship (ER) Model: Entity Types, Entity Sets, Attributes, Keys, Relationship Types, Relationship Sets, Roles and Structural Constraints, Weak Entity Types, ER Naming Conventions. Enhanced Entity-Relationship (EER) Model. |
| **UNIT-II: 10hrs** |
| Database Design Theory and Normalization: Functional Dependencies, Normal Forms based on Primary Keys, Second and third Normal Forms, Boyce-Codd Normal Form, Multivalued Dependency and Fourth Normal Form, Join Dependencies and Fifth Normal Form. |
| **UNIT-III: 10hrs** |
| Relational data Model and SQL: Relational Model Concepts, Basic SQLs, SQL Data Definition and Data types, Constraints in SQL, Retrieval Queries in SQL, INSERT, DELETE, UPDATE Statements in SQL, Relational Algebra and Relational Calculus: Unary Relational Operations: SELECT and PROJECT, Binary Relation: JOIN and DIVISION. |
| **UNIT-IV: 10hrs** |
| Managing Console I/O Operations: C++ Streams, C++ Stream Classes, Unformatted I/O Introduction to Transaction Processing Concepts and Theory: Introduction to Transaction Processing, Transaction and System Concepts, Properties of Transactions, Recoverability, Serializability, Concurrency Control Techniques, Locking techniques for Concurrency Control, Concurrency Control based on Time-Stamp Ordering. |
| **Text Books:** |
| 1. Ramez Elmasri, Shamkant B. Navathe, "Fundamentals of Database Systems", Pearson Education , Seventh Edition, 2015. |
| **Reference Books:** |
| 1. C. J. Date, "An Introduction to Database System", Pearson Education, Eighth Edition, 2005 |

| | **COURSE OUTCOMES:** After completion of this course successfully, the students will be able to- |
|---|---|
| CO1 | Understand the benefits and applications of database systems. |
| CO2 | Learn the aspects of basic designing of a database. |
| CO3 | Identify the functional requirements of a database and analyse its quality. |
| CO4 | Design a database for a small enterprise entertaining the security and concurrency control mechanisms. |

**CORE–6 Practical: Database Systems Lab (Credits: 2)**
**Marks:25**

Create and use the following database schema to answer the given queries.

**EMPLOYEE Schema**

| Field | Type | NULL | KEY | DEFAULT |
|---|---|---|---|---|
| Eno | Char(3) | NO | PRI | NIL |
| Ename | Varchar(50) | NO | | NIL |
| Job_type | Varchar(50) | NO | | NIL |
| Manager | Char(3) | Yes | FK | NIL |

| | | | | |
|---|---|---|---|---|
| Hire_date | Date | NO | | NIL |
| Dno | Integer | YES | FK | NIL |
| Commission | Decimal(10,2) | YES | | NIL |
| Salary | Decimal(7,2) | NO | | NIL |

**DEPARTMENT Schema**

| Field | Type | NULL | KEY | DEFAULT |
|---|---|---|---|---|
| Dno | Integer | No | PRI | NULL |
| Dname | Varchar(50) | Yes | | NULL |
| Location | Varchar(50) | Yes | | New Delhi |

**Query List**

1. Query to display Employee Name, Job, Hire Date, Employee Number; for each employee with the Employee Number appearing first.
2. Query to display unique Jobs from the Employee Table.
3. Query to display the Employee Name concatenated by a Job separated by a comma.
4. Query to display all the data from the Employee Table. Separate each Column by a comma and name the said column as THE_OUTPUT.
5. Query to display the Employee Name and Salary of all the employees earning more than $2850.
6. Query to display Employee Name and Department Number for the Employee No= 7900.
7. Query to display Employee Name and Salary for all employees whose salary is not in the range of $1500 and $2850.
8. Query to display Employee Name and Department No. of all the employees in Dept 10 and Dept 30 in the alphabetical order by name.
9. Query to display Name and Hire Date of every Employee who was hired in 1981.
10. Query to display Name and Job of all employees who don't have a current Manager.
11. Query to display the Name, Salary and Commission for all the employees who earn commission.
12. Sort the data in descending order of Salary and Commission.
13. Query to display Name of all the employees where the third letter of their name is 'A'.
14. Query to display Name of all employees either have two 'R's or have two 'A's in their name and are either in Dept No = 30 or their Mangers Employee No = 7788.
15. Query to display Name, Salary and Commission for all employees whose Commission Amount is 14 greater than their Salary increased by 5%.
16. Query to display the Current Date.
17. Query to display Name, Hire Date and Salary Review Date which is the 1st Monday after six months of employment.
18. Query to display Name and calculate the number of months between today and the date each employee was hired.
19. Query to display the following for each employee <E-Name> earns < Salary> monthly but wants <3*Current Salary>. Label the Column as Dream Salary.
20. Query to display Name with the 1st letter capitalized and all other letter lower case and length of their name of all the employees whose name starts with 'J', 'A' and 'M'.
21. Query to display Name, Hire Date and Day of the week on which the employee started.
22. Query to display Name, Department Name and Department No for all the employees.
23. Query to display Unique Listing of all Jobs that are in Department # 30.
24. Query to display Name, Department Name of all employees who have an 'A' in their name.

25. Query to display Name, Job, Department No. and Department Name for all the employees working at the Dallas location.
26. Query to display Name and Employee no. Along with their Manger's Name and the Manager's employee no; along with the Employees Name who do not have a Manager.
27. Query to display Name, Department No. And Salary of any employee whose department No. and salary matches both the department no. And the salary of any employee who earns a commission.
28. Query to display Name and Salaries represented by asterisks, where each asterisk (*) signifies $100.
29. Query to display the Highest, Lowest, Sum and Average Salaries of all the employees.
30. Query to display the number of employees performing the same Job type functions.
31. Query to display the no. of managers without listing their names.
32. Query to display the Department Name, Location Name, No. of Employees and the average salary for all employees in that department.
33. Query to display Name and Hire Date for all employees in the same dept. as Blake.
34. Query to display the Employee No. And Name for all employees who earn more than the average salary.
35. Query to display Employee Number and Name for all employees who work in a department with any employee whose name contains a 'T'.
36. Query to display the names and salaries of all employees who report to King.
37. Query to display the department no, name and job for all employees in the Sales department.

## CC-7: DISCRETE MATHEMATICAL STRUCTURES

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

### COURSE OVERVIEW:

This is an introductory course in discrete mathematics. The goal of this course is to introduce students to ideas and techniques from discrete mathematics that are widely used in science and engineering. This course teaches the students techniques in how to think logically and mathematically and apply these techniques in solving problems. To achieve this goal, students will learn logic and proof, sets, functions, as well as algorithms and mathematical reasoning. Key topics involving relations, graphs, trees, and formal languages and computability are covered in this course.

### COURSE OBJECTIVES:

1. To learn the mathematical foundations for Computer Science.
2. Topics covered essential for understanding various courses.
3. Understand and construct mathematical arguments
4. Develop recursive algorithms based on mathematical induction
5. Know essential concepts in graph theory and related algorithms
6. Apply knowledge about discrete mathematics in problem solving

| Core Course-7 | |
|---|---|
| **Discrete Mathematical Structures** | |
| **UNIT-I:** | **12 hrs** |
| Logics and Proof: Propositional Logic, Propositional Equivalences, Predicates and Quantifiers Nested Quantifiers, Rules inference, Mathematical Induction. Sets and Functions: Sets, Relations, Functions, Closures of Equivalence Relations, Partial ordering well ordering, Lattice, Sum of products and product of sums principle of Inclusions and Exclusions | |
| **UNIT-II:** | **08hrs** |
| Combinatory: Permutations, Combinations, Pigeonhole principle Recurrence Relation: Linear and Non-linear Recurrence Relations, Solving Recurrence Relation using Generating Functions. | |
| **UNIT-III:** | **10hrs** |
| Graphs: Introduction to graphs, graphs terminologies, Representation of graphs, Isomorphism, Connectivity & Paths: Connectivity, Euler and Hamiltonian Paths, Introduction to tree, tree traversals, spanning tree and tree search: Breadth first search, Depth first search, cut-set, cut-vertex. | |
| **UNIT-IV:** | **10hrs** |
| Modeling Computation: Finite State Machine, Deterministic Finite Automata (DFA), Non-Deterministic Finite Automata (NFA), Grammars and Language, Application of Pumping Lemma for Regular Language. | |
| **Text Book:** | |
| 1. Kenneth H. Rosen, "Discrete Mathematics and its Applications with Combinatory and Graph Theory", 7th edition. | |
| **Reference Books:** | |
| 1. C. L. Liu and D.P. Mohapatra, "Elements of Discrete Mathematics", TMH, 2012 | |
| 2. J. P Tremblay, R. Manohar, "Discrete Mathematical Structures with Applications to Computer Science", TMH, 1997 | |
| 1. J. K. Mantri & T. K Tripathy, "A Modern Approach to Discrete Mathematics and Structure", Laxmi Publication. | |

| | **COURSE OUTCOMES:**<br>After completion of this course successfully, the students will be able to- |
|---|---|
| **CO1** | Able to use logical notation to define and reason about fundamental mathematical concepts such as sets, relations, and functions. |
| **CO2** | To apply mathematical foundations, algorithmic principles, and computer science theory to the modelling and design of computer based systems. |
| **CO3** | Able to construct simple mathematical proofs and possess the ability to verify them. |
| **CO4** | Model problems in Computer Science using graphs and trees methods. |
| **CO5** | To Understand and prove fundamental results and solve algebraic problems using appropriate techniques |

**CORE COURSE– 7 Practical: Discrete Mathematical Structure Lab (Credits: 2)**
**Marks: 25**

Write the following programs using C/ C++

1. Tower of Hanoi
2. Graph representation using Adjacency List.
3. Graph representation using Adjacency Matrix.
4. String Matching using finite state machine.
5. Detecting whether a number is even or odd using Finite State Machine.
6. To identify keywords such as char, const, continue using Finite State Machine.
7. To find the power set for a given set.
8. To find GCD of two numbers using recursion.
9. To find Binomial coefficients.
10. To find Permutation and Combination result for a given pair of values n and r.
11. To check a number is prime or not.
12. To calculate the Euclidean distance between two points.
13. To find the Roots of polynomials.
14. Find the shortest path pair in a plane.

## Semester-IV

# CC-8: OPERATING SYSTEMS

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

### COURSE OVERVIEW:

This course will introduce the core concepts of operating systems, such as processes and threads, scheduling, synchronization, dedalock, memory management and file systems. The goal of the programming assignments is to give students some exposure to operating system code.

### COURSE OBJECTIVES:

1. To understand Operating system structure and services.
2. To make aware of different types of Operating System and their services.
3. To understand the concept of a Process, memory, storage and I/O management.

| **Core Course-8** |
|---|
| **Operating Systems** |
| **UNIT-I:** **8 hrs** |
| Introduction to Operating System, System Structures: Operating system services, system calls, system programs, Operating system design and implementation, Operating system structure. |

| UNIT-II: | 10hrs |
|---|---|
| Process Management: Process Concept, Operations on processes, Process scheduling and algorithms, Inter-process Communication, Concepts on Thread and Process, Deadlocks: Deadlock detection, deadlock prevention, and deadlock avoidance fundamentals. | |
| **UNIT-III:** | **12hrs** |
| Memory Management Strategies: Swapping, Contiguous Memory Allocation, Paging, Segmentation, Virtual Memory Management: Concepts, implementation (Demand Paging), Page Replacement, Thrashing | |
| **UNIT-IV:** | **10hrs** |
| Storage Management: File System concept, Access Methods, File System Mounting, File Sharing and File Protection, Implementing File Systems, Kernel I/O Systems | |
| **Text Book:** | |
| **1.** Abraham Silberschatz, Peter B. Galvin, and Greg Gagne,"Operating System Concepts", Wiley Student Edition ,Eighth Edition, 2009. | |
| **Reference Books:** | |
| 1.Tanenbaum, "Modern Operating Systems" , Pearson , Fifth Edition, 2022. | |
| 2. Richard Fox, "Linux with Operating System Concepts", CRC Press, Second Edition,2014 | |
| 3. Richard Blum ,"Linux Command Line and Shell Scripting Bible", Wiley Publishing, Third Edition,2015. | |

| | **COURSE OUTCOMES:**<br>After completion of this course successfully, the students will be able to- |
|---|---|
| **CO1** | Understands the different services provided by Operating System at different level. |
| **CO2** | Describe, contrast and compare differing structures of operating systems. |
| **CO3** | Understands the use of different process scheduling algorithms, memory management Algorithms and synchronization techniques to avoid deadlock. |
| **CO4** | Gain practical knowledge of how programming languages, operating systems, and architectures interact and how to use each one effectively |

**CORE COURSE–8 Practical: Operating Systems Lab (Credits: 2)**
**Marks:25**

1. Write a program (using fork() and/or exec() commands) where parent and child execute:
a) same program, same code. b) same program, different code. c) before terminating, the parent waits for the child to finish its task.
2. Write a program to report behavior of Linux kernel including kernel version, CPU type and model. (CPU information)
3. Write a program to report behavior of Linux kernel including information on configured memory, amount of free and used memory. (memory information)

4. Write a program to print file details including owner access permissions, file access time, where file name is given as argument.
5. Write a program to copy files using system calls.
6. Write a program using C to implement FCFS scheduling algorithm.
7. Write a program using C to implement Round Robin scheduling algorithm.
8. Write a program using C to implement SJF scheduling algorithm.
9. Write a program using C to implement non-preemptive priority based scheduling algorithm.
10. Write a program using C to implement preemptive priority based scheduling algorithm.
11. Write a program using C to implement SRTF scheduling algorithm.
12. Write a program using C to implement first-fit, best-fit and worst-fit allocation strategies.

## CC-9: COMPUTER NETWORKS

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

### COURSE OVERVIEW:

The course introduces main concepts of networking; application areas; classification; reference models; transmission environment; technologies; routing algorithms; IP, UDP and TCP protocols; reliable data transferring methods; application protocols; network security; management systems; perspectives of communication networks.

### COURSE OBJECTIVES:

1. The objective of the course is to equip the students with a general overview of the concepts and fundamentals of computer networks.
2. Familiarize the students with the standard models for the layered approach to communication between machines in a network and the protocols of the various layers.
3. Introducing fundamentals of networking concepts with the help of layered architecture which includes OSI and TCP/IP model.

| Core Course-9 | |
|---|---|
| **Computer Networks** | |
| **UNIT-I:** | **10hrs** |
| Introduction to Data Communications and Network Models: Protocols and Standards, Layers in OSI Models, Analog and Digital Signals, Transmission Modes, Transmission Impairment, Data Rate Limits, Performance, Digital Transmission, Network Devices & Drivers: Router, Modem, Repeater, Hub, Switch, Bridge (fundamental concepts only). | |
| **UNIT-II:** | **10hrs** |
| Signal Conversion: Digital-to-Digital Conversion, Analog-to-Digital Conversion, Digital-to- analog Conversion, Analog-to-analog Conversion. Transmission Media: Guided Media, Unguided Media, Switching Techniques: Packet Switching, Circuit Switching, Datagram Networks, Virtual-Circuit | |

| |
|---|
| Networks, and Structure of a Switch |
| **UNIT-III:** **10hrs** |
| Error Detection and Correction: Checksum, CRC, Data Link Control: Framing, Flow and Error Control, Noiseless Channels, Noisy channels, (Stop and Wait ARQ, Slidding Window Protocol, Go Back N, Selective Repeat) HDLC, Point-to-Point Protocol. Access Control: TDM, CSMA/CD, and Channelization (FDMA, TDMA, and CDMA). |
| **UNIT-IV:** **10hrs** |
| Network Layer: Logical Addressing, IPv4 Addresses, IPv6 Addresses, Virtual-Circuit Networks: Frame Relay and ATM, Transport Layer: Process-Process Delivery: UDP, TCP. Application layers: DNS, SMTP, POP, FTP, HTTP, Basics of WiFi (Fundamental concepts only), Network Security: Authentication, Basics of Public Key and Private Key, Digital Signatures and Certificates (Fundamental concepts only). |
| **Text Book:** |
| **1.** Behrouz A. Forouzan, "Data Communications and Networking", TMH, Fifth Edition, 2013. |
| **Reference Books:** |
| 1. A.S.Tanenbaum, "Computer Networks", Pearson Education, Fourth Edition, 2018<br>2. S. Keshav, "An Engineering Approach to Computer Networks", Pearson Education, Second Edition, 1997<br>3. Kurose, Ross, "Computer Networking: A top down approach", Pearson Education, Sixth Edition, 2017 |

| | **COURSE OUTCOMES:**<br>After completion of this course successfully, the students will be able to- |
|---|---|
| **CO1** | Gain the knowledge of the basic computer network technology. |
| **CO2** | Gain the knowledge of the functions of each layer in the OSI and TCP/IP reference model. |
| **CO3** | Obtain the skills of subnetting and routing mechanisms. |
| **CO4** | Familiarity with the essential protocols of computer networks, and how they can be applied in network design and implementation. |

**CORE COURSE–9 Practical: Computer Networks Lab (Credits: 2)**
**Marks:25**
**Use C/C++/ any Network Simulator**

1. Simulate Even Parity generator and checker.
2. Simulate two dimensional Parity generator and checker.
3. Simulate check sum generator and checker.
4. Simulate Hamming code method.
5. Simulate Cyclic Redundancy Check (CRC) error detection algorithm for noisy channel.
6. Simulate and implement stop and wait protocol for noisy channel.
7. Simulate and implement go back n sliding window protocol.
8. Simulate and implements elective repeat sliding window protocol.
9. Simulate and implement distance vector routing algorithm.

# CC-10: COMPUTER GRAPHICS

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

## COURSE OVERVIEW:

The course introduces computer modeling of 2D & 3-D objects and efficiently generating photorealistic renderings on color raster graphics devices.

## COURSE OBJECTIVES:

1. To be able to learn the core concepts of Computer Graphics.
2. To be able to create effective programs for solving graphics problems.
3. To be able to generate photorealistic renderings on color raster graphics devices.

| Core Course-10 | |
|---|---|
| **Computer Graphics** | |
| **UNIT-I:** | **10hrs** |
| Computer Graphics: A Survey of Computer graphics, Overview of Graphics System: Video Display Devices, Raster-Scan Systems, Input Devices, Hard-Copy Devices, Graphics Software. | |
| **UNIT-II:** | **10hrs** |
| Graphics Output Primitives: Point and Lines, Algorithms for line, circle & ellipse generation, Filled-Area Primitives. Attributes of Graphics Primitives: Point, line, curve attributes, fill area attributes, Fill methods for areas with irregular boundaries | |
| **UNIT-III:** | **10hrs** |
| Geometric Transformations (both2-D&3-D): Basic Geometric Transformations, Transformation Matrix, Types of transformation in 2-D and 3-D Graphics: Scaling, Reflection, shear transformation, rotation, translation. 2-D, 3-D transformation using homogeneous coordinates. | |
| **UNIT-IV:** | **10hrs** |
| Two Dimensional Viewing: Introduction to viewing and clipping, Viewing transformation in 2-D, Viewing pipeline, Clipping Window, Clipping Algorithms: Point clipping, Line clipping and Polygon clipping. | |
| **Text Book:** | |
| 1.D. Hearn and P. Baker, Computer Graphics – C Version, Pearson Education, 2nd Edition. 2004.<br>2.F. S. Hill, Computer Graphics using OpenGL, Pearson Education, 2nd Edition, 2003. | |
| **Reference Books:** | |
| 1.J. F. Huges, A. V. Dam, M. McGuire, D. F. Sklar, J. D. Foley, S. K. Feiner, and K. Akeley, Computer Graphics: Principles and Practice, Addison-Wesley Professional, 3rd Edition, 2013.<br>2.D. Hearn, M. P. Baker and W. Caritthers, Computer Graphics with OpenGL, Prentice Hall India, 4th Edition, 2010.<br>3.S. Harrington, Computer Graphics - A Programming Approach, Tata McGraw-Hill, 2nd Edition, 2004. | |

| | COURSE OUTCOMES:<br>After completion of this course successfully, the students will be able to- |
|---|---|
| CO1 | Describe the basics of computer graphics and its applications. |
| CO2 | Explore the standard line, circle, and area filling algorithms. |
| CO3 | Design various transformation models in 2D spaces |
| CO4 | Design various transformation models in 3Dspace |

**CORE COURSE–10 Practical: Computer Graphics Lab (Credits: 2)**
**Marks:25**

1. Write a program to implement Bresenham's line drawing algorithm.
2. Write a program to implement mid-point circle drawing algorithm.
3. Write a program to clip a line using Cohen and Sutherland line clipping algorithm
4. Write a program to clip a polygon using Sutherland Hodgeman algorithm.
5. Write a program to fill a polygon using Scan line fill algorithm.
6. Write a program to apply various 2D transformations on a 2D object (use homogenous coordinates).
7. Write a program to apply various 3D transformations on a 3D object and then apply parallel and perspective projection on it.

## Semester-V

## CORE COURSE-11: WEB TECHNOLOGY

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

### COURSE OVERVIEW:

This course provides an overview of web server and clients. It includes the elements required to design web pages both passive and dynamic. It helps the learners in knowing about the active web pages and incorporating client side validations. The course makes the students learn to create server side database and connect the database to a web application for retrieval of information.

### COURSE OBJECTIVES:

1. To learn the fundamentals of web application design.
2. To design and develop standard and interactive web pages.
3. To learn some popular web scripting languages.

| Core-11 | |
|---|---|
| **Web Technology** | |
| **UNIT-I:** | **10hrs** |
| Web Essentials: Clients, Servers and Communication: The Internet – Basic Internet protocols – The WWW, HTTP request message – response message, web clients web servers – case study.<br>Introduction to HTML: HTML, HTML domains, basic structure of an HTML document – creating an HTML document, mark up tags, heading, paragraphs, line breaks, HTML tags. Elements of HTML, working with text, lists, tables and frames, working with hyperlink, images<br>and multimedia, forms and controls. | |

| UNIT-II: | 10hrs |
|---|---|
| Introduction to cascading style sheets: Concepts of CSS, creating style sheet, CSS properties, CSS styling (background, text format, controlling fonts), working with the block elements and objects. Working who lists and tables, CSS ID and class. Box model (introduction, border properties, padding properties, margin properties), CSS colour, groping, Dimensions, display, positioning, floating, align, pseudo class, Navigation bar, image sprites. | |
| **UNIT-III:** | **10hrs** |
| Java scripts: Client side scripting, what is java script, simple java script, variables, functions, conditions, loops and repetitions. Java scripts and objects, java script own objects, the DOM and web browser environment, forms and validations.<br>DHTML: Combining HTML, CSS, java scripts, events and buttons, controlling your browser. | |
| **UNIT-IV:** | **10hrs** |
| PHP: Starting to script on server side, PHP basics, variables, data types, operators, expressions, constants, decisions and loop making decisions. Strings – creating, accessing strings, searching, replacing and formatting strings. Arrays: Creation, accessing array, multidimensional arrays, PHP with Database. | |
| **Text Books:** | |
| 1. Black Book,"Web Technologies ", DreamTech Press, First Edition, 2009.<br>2. Matt Doyle, "Beginning PHP 5.3". wrox-Willey publishing, First Edition, 2009.<br>3. John Duckett, "Beginning HTML, XHTML, CSS and Java script", First Edition, 2010. | |
| **Reference Books:** | |
| 1. Steven M. Schafer," HTML, XHTML and CSS Bible", Willey India, Fifth Edition, 2010. | |

| | **COURSE OUTCOMES:**<br>After completion of this course successfully, the students will be able to- |
|---|---|
| **CO1** | Understand the programming elements to develop web application. |
| **CO2** | Learn to develop the solutions to complex problems using appropriate method and technology. |
| **CO3** | Identify the functional requirements of a web application and analyse it to choose sustainable web development methodology. |
| **CO4** | Develop web based application using suitable client side and server side web technologies. |

**CORE–11 Practical: Web Technology Lab (Credits: 2)**
**Marks:25**

1. Acquaintance with elements, tags and basic structure of HTML files.
2. Practicing basic and advanced text for formatting.
3. Practice use of image, video and sound in HTML documents.
4. Designing of web pages- Document layout, list, tables.
5. Practicing Hyperlink of web pages, working with frames.
6. Working with forms and controls.
7. Acquaintance with creating style sheet, CSS properties and styling.
8. Working with background, text, font, list properties.
9. Working with HTML elements box properties in CSS.

10. Develop simple calculator for addition, subtraction, multiplication and division operation using java script.
11. Create HTML page with java script which takes integer number as a input and tells whether the number is odd or even.
12. Create HTML page that contains form with fields name, Email, mobile number, gender, favourite colour and button; now write a java script code to validate each entry. Also write a code to combine and display the information in text box when button is clicked.
13. Write a PHP program to check if number is prime or not.
14. Write a PHP program to print first ten Fibonacci numbers.
15. Create a MySQL data base and connect with PHP.
16. Write PHP script for string and retrieving user information from my SQL table.
a. Write a HTML page which takes Name, Address, Email and Mobile number from user (register PHP).
b. Store this data in MySQL data base.
c. Next page display all user in HTML table using PHP (display .PHP).
17. Using HTML, CSS, Javascript, PHP, MySQL, design a authentication module of a web page.

## CORE – 12: SOFTWARE ENGINEERING

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

### COURSE OVERVIEW:

It focuses on key Computer Programming and Software System concepts through the study of subjects such as Physics, Chemistry, Mathematics, and Data Structures and includes summer projects & internships.

### COURSE OBJECTIVES:

1. To learn the way of developing software with high quality and the relevant techniques.
2. To introduce software engineering principles for industry standard.
3. To focus on Project management domain and Software risks management.

| CORE-12 | |
|---|---|
| **Software Engineering** | |
| **UNIT-I:** | **10hrs** |
| Introduction: Evolution of Software to an Engineering Discipline, Software Development Projects, Exploratory Style of Software Development, Emergence of Software Engineering, Changes in Software Development Practices, Computer Systems Engineering.<br>Software Lifecycle Models: Waterfall Model and its Extensions, Rapid Application Development (RAD), Agile Development Models, Spiral Model. | |
| **UNIT-II:** | **10hrs** |
| Software Project Management: Software Project Management Complexities, Responsibilities of a | |

Software Project Manager, Project Planning, Metrics for Project Size Estimation, Project Estimation Techniques, Empirical Estimation Techniques, COCOMO, Halstead's Software Science, Staffing Level Estimation, Scheduling, Organization and Team Structures, Staffing, Risk Management, Software Configuration Management.

**UNIT-III:** **10hrs**

Requirement Analysis and Specification: Requirements Gathering and Analysis, Software Requirement Specifications, Formal System Specification Axiomatic Specification, Algebraic Specification, Executable Specification and 4GL.

Software Design: Design Process, Characterize a Good Software Design, Cohesion and Coupling, Layered Arrangements of Modules, Approaches to Software Design (Function Oriented & Object-Oriented).

**UNIT-IV:** **10hrs**

Coding and Testing: Coding: Code Review, Software Documentation, Testing, Unit Testing, Black Box and White Box Testing, Debugging, Program Analysis Tools, Integration Testing, System Testing, Software Maintenance

**Text Books:**

1. Fundamental of Software Engineering, Rajib Mall, PHI Publication, India, Fifth Edition.

**Reference Books:**

1. Software Engineering– Ian Sommerville, Pearson,Tenth Edition.
2. Software Engineering Concepts and Practice – Ugrasen Suman, Cengage Learning India Pvt, Ltd.
3. R. Misra, C. Panigrahi, B. Panda: Principles of Software Engineering & System Design, YesDee Publication

| | **COURSE OUTCOMES:**<br>After completion of this course successfully, the students will be able to- |
|---|---|
| **CO1** | How to apply the software engineering lifecycle by demonstrating competence in communication, planning, analysis, design, construction, and deployment |
| **CO2** | An ability to work in one or more significant application domains |
| **CO3** | Work as an individual and as part of a multidisciplinary team to develop and deliver quality software |
| **CO4** | Demonstrate an understanding of and apply current theories, models, and techniques that provide a basis for the software lifecycle |

**CORE – 12 Practical: Software Engineering Lab (Credit: 2)**
**Marks:25**

**S. No. Practical Title**

1. • Problem Statement,
    - Process Model
    - Reuirement Analysis:Creating a Data Flow
    - Data Dictionary, Use Cases
2. Project Management:
    - Computing FP
    - Effort
    - Schedule, Risk Table, Timeline chart
3. Design Engineering:
    - Architectural Design
    - Data Design, Component Level Design
4. Testing:
    - Basis Path Testing

## *Sample Projects:*

1. **Criminal Record Management:** Implement a criminal record management system for jailers, police officers and CBI officers.
2. **Route Information**: Online information about the bus routes and their frequency and fares
3. **Car Pooling**: To maintain a web based intranet application that enables the corporate employees within an organization to avail the facility of carpooling effectively.
4. Patient Appointment and Prescription Management System
5. Organized Retail Shopping Management Software
6. Online Hotel Reservation Service System
7. Examination and Result computation system
8. Automatic Internal Assessment System
9. Parking Allocation System
10. Wholesale Management System

**Semester-VI**

# CC-13: ARTIFICIAL INTELLIGENCE

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

## COURSE OVERVIEW:

The course addresses the basic concepts of AI principles and approaches. It teaches the basic principles of AI in solutions that require problem solving, inference, perception, knowledge representation, and learning. This course forms an ability to share in discussions of AI, its current scope and limitations, and societal implications.

## COURSE OBJECTIVES:

1. To impart knowledge about Artificial Intelligence.
2. To give understanding of the main abstractions and reasoning for intelligent systems.
3. To enable the students to understand the basic principles of Artificial Intelligence in various applications.

| **Core Course-13** | |
|---|---|
| **Artificial Intelligence** | |
| **UNIT-I:** | **8hrs** |
| Introduction to Artificial Intelligence, Background and Applications, Turing Test and Rational Agent approaches to AI, Introduction to Intelligent Agents, their structure, behavior and environment | |
| **UNIT-II:** | **12hrs** |
| Problem Solving and Searching Techniques: Problem Characteristics, Production Systems, Control Strategies, Breadth First Search, Depth First Search, Hill climbing and its Variations, Heuristics Search Techniques: Best First Search, A* algorithm, Constraint Satisfaction Problem, Introduction to Game Playing, Min-Max and Alpha-Beta pruning algorithms. | |
| **UNIT-III:** | **10hrs** |
| Knowledge Representation : Introduction to First Order Predicate Logic, Resolution Principle, Unification, Semantic Nets, Conceptual Dependencies, Frames, and Scripts, Production Rules, Conceptual Graphs. | |
| **UNIT-IV:** | **10hrs** |
| Dealing with Uncertainty and Inconsistencies Truth Maintenance System, Default Reasoning, Probabilistic Reasoning, Bayesian Probabilistic Inference, Possible World Representations, Basics of NLP. | |
| **Text Books:** | |
| **1.** Stuart Russell and Peter Norvig, “Artificial Intelligence a Modern Approach”, Pearson Education, Third Edition, 2010. | |
| **Reference Books:** | |
| 1.Eugene Charniak and Drew McDermott, “Introduction to Artificial Intelligence”, Pearson Education, Second Edition, 2005.<br>2. G. Luger, “Artificial Intelligence: Structures and Strategies for complex problem solving”, Pearson Education, Fourth Edition, 2008.<br>3. Gerhard Weiss, “Multi Agent Systems”, MIT Press, Second Edition, 2013. | |

| | COURSE OUTCOMES:<br>After completion of this course successfully, the students will be able to- |
|---|---|
| CO1 | Solve basic AI based problems. |
| CO2 | Define the concept of Artificial Intelligence. |
| CO3 | Apply AI techniques to real-world problems to develop intelligent systems. |
| CO4 | Select appropriately from a range of techniques when implementing intelligent systems. |

**CORE COURSE–13 Practical: Artificial Intelligence Lab (Credits: 2)**
**Marks:25**

Write a Prolog program

1. To find the factorial of a number.
2. To remove the nth item from a list.
3. To find the permutation of a set.
4. To implement append for two lists.
5. To implement palindrome.
6. To find the greater of two numbers X and Y.
7. To find the greatest number in the list of numbers.
8. To find the sum of given list of numbers.
9. To find the reverse of a list.
10. To solve 8 queens problem.
11. To solve 8-puzzle problem using best first search.
12. To implement DFS.
13. To implement BFS.
14. To implement best first search.
15. To solve traveling salesman problem.

## CC-14: ALGORITHM DESIGN TECHNIQUES

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

### COURSE OVERVIEW:

Algorithms are the soul of computing .This course introduces basic methods for the design and analysis of efficient algorithms emphasizing methods useful in practice. Different algorithms for a given computational task are presented and their relative merits evaluated based on performance measures

### COURSE OBJECTIVES:

1. To be able to learn design principles and concepts of algorithms.

2. To have a mathematical foundation in analysis of algorithms.
3. Demonstrate a familiarity with major algorithms and data structures.

| **Core Course-14** | |
|---|---|
| **Algorithm Design Techniques** | |
| **UNIT-I:** | **12hrs** |
| **Introduction**: Algorithm specification: Pseudo code, Space complexity and time complexity, Analysis and design of Insertion sort algorithm, Divide and Conquer paradigm, Recurrence relations, Solving Recurrences: Substitution methods, Recursion tree method, and Master method | |
| **UNIT-II:** | **8hrs** |
| **Searching and Sorting**: Analysis of Linear Search, Binary Search, Merge Sort and Quick Sort, Heap Sort. Hashing: Hash functions, Hash table, and Collision resolution: Chaining and Open Addressing (Linear probing, Quadratic probing, Double hashing). | |
| **UNIT-III:** | **10hrs** |
| **Greedy Technique**: General Method, Applications: Fractional Knapsack Problem, Job Sequencing with Deadlines, Huffman Codes.<br>**Dynamic Programming**: General Method, Applications: Matrix Chain Multiplication, Longest common subsequence. | |
| **UNIT-IV:** | **10hrs** |
| **Graph Algorithms**: Representations of Graphs, Breadth-first search, Depth-first search, Topological sort, Minimum Spanning Trees: Prim‘s and Kruskal‘s algorithm, Single-source shortest paths: Bellman-Ford algorithm, Dijkstra‘s algorithm. | |
| **Text Book:** | |
| 1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein,” Introduction to Algorithms.”, MIT Press , Fourth Edition, 2022. | |
| **Reference Book:** | |
| 1.Jon Kleinberg, Eva Tardos,”Algorithm Design”,Pearson Education,First Edition.2006 | |

| | **COURSE OUTCOMES:**<br>After completion of this course successfully, the students will be able to- |
|---|---|
| **CO1** | Understand the basic notation for analyzing the performance of the algorithms. |
| **CO2** | Ability to analyze asymptotic runtime complexity of various algorithms including formulating recurrence relations. |
| **CO3** | Ability to understand and design algorithms using greedy strategy, divide and conquer technique, dynamic programming approaches to solve various problems |
| **CO4** | Able to Compare between different data structures and pick an appropriate data structure for a design situation. |

**CORE COURSE–14 Practical: Algorithm Design Techniques Lab (Credits: 2)**
**Marks:25**

**Using C or C++ implement the following**

1. Quick sort.
2. Heap sort.
3. Merge sort.
4. Matrix Multiplication using recursion.
5. Linear Search.
6. Binary Search.
7. Huffman code.
8. Fractional knapsack problem.
9. Matrix chain multiplication.
10. Longest Common Subsequence.
11. Prim's algorithm.
12. Kruskal's algorithm.
13. BFS.
14. DFS.
15. Dijkstra Algorithm

# Discipline Specific Electives

**Semester-V**

## DSE-1: NUMERICAL TECHNIQUES

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

**COURSE OVERVIEW:**

Numerical analysis finds application in all fields of engineering and the physical sciences, and in the 21st century also the life and social sciences, medicine, business and even the arts. Employability includes forecasting and predicting in the field of machine learning. Solving differential and integral equations.

**COURSE OBJECTIVES:**

1. To learn various numerical techniques.
2. To be able to implement different numerical techniques using programming language.
3. To be able to solve numerical techniques related problems.

| **DSE-1** | |
|---|---|
| **Numerical Techniques** | |
| **UNIT-I:** | **10hrs** |
| Floating point representation and computer arithmetic, Significant digits, Errors: Round-off error, Local truncation error, Global truncation error, Order of a method, Convergence and terminal | |

| |
|---|
| conditions, efficient computations. |
| **UNIT-II:** **10hrs** |
| Bi section method, Secant method, Regula−Falsi method Newton−Raphson method, Newton‘s method for solving nonlinear systems. |
| **UNIT-III:** **10hrs** |
| Interpolation: Lagrange‘s form and Newton‘s form Finite difference operators, Gregory Newton forward and backward differences Interpolation Piecewise polynomial interpolation: Linear interpolation. |
| **UNIT-IV:** **10hrs** |
| Numerical integration: Trape zoid rule, Simpson‘s rule (only method),Newton−Cotes formulas, Gaussian quadrature, Ordinary differential equation: Euler‘s method Modified Euler‘s methods, Runge-Kutta second methods |
| **Text Book:** |
| 1. S. S. Sastry, “Introductory Methods of Numerical Analysis” PHI learning, 5th Edition 2012.<br>2. M. K. Jain,S. R. K. Iyengar and R. K. Jain, “Numerical Methods for Scientific and Engineering Computation”, New Age International Publisher, 6th Edition, 2012. |
| **Reference Books:** |
| 1. Josef Stoer and Roland Bulirsch, “Introduction to Numerical Analysis”, Springer, 2nd Edition, 1996. |

| | **COURSE OUTCOMES:**<br>After completion of this course successfully, the students will be able to- |
|---|---|
| **CO1** | Describe Understanding and Learning of numerical methods for numerical analysis |
| **CO2** | Describe Learning of tracing errors in Numerical methods and analyze and predict it. |
| **CO3** | Apply Learning of application of Statistical methods. |
| **CO4** | Discuss concepts of numerical methods used for different applications. |

**DSE–1 Practical: Numerical Techniques Lab (Credits: 2)**
**Marks:25**

1. Find the roots of the equation by bisection method.
2. Find the roots of the equation by secant/Regula−Falsi method.
3. Find the roots of the equation by Newton’s method.
4. Find the solution of a system of nonlinear equation using Newton’s method.
5. Find the solution of tri-diagonal system using Gauss Thomas method.
6. Find the solution of system of equations using Jacobi/Gauss-Seidel method.
7. Find the cubic spline interpolating function.
8. Evaluate the approximate value of finite integrals using Gaussian/Romberg integration.
9. Solve the boundary value problem using finite difference method.

# DSE – 2: UNIX SHELL PROGRAMMING

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

## COURSE OVERVIEW:

The shell is the operating system's command-line interface (CLI) and interpreter for the set of commands that are used to communicate with the system. A shell script is usually created for command sequences in which a user has a need to use repeatedly in order to save time.

## COURSE OBJECTIVES:

1. To learn the basics of UNIX OS, UNIX commands and File system.
2. To familiarize students with the Linux environment.
3. To learn fundamentals of shell scripting and shell programming.
4. To be able to write simple programs using UNIX.

| **DSE – 2** | |
|---|---|
| **Unix Shell Programming** | |
| **UNIT-I:** | **10hrs** |
| **Introduction:** Unix Operating systems, Difference between Unix and other operating systems, Features and Architecture, Installation, Booting and shutdown process, System processes (an overview), External and internal commands, Creation of partitions in OS, Processes and its creation phases – Fork, Exec, wait, exit. | |
| **UNIT-II:** | **10hrs** |
| User Management and the File System: Types of Users, Creating users, Granting rights, User management commands, File quota and various file systems available, File System Management and Layout, File permissions, Login process, Managing Disk Quotas, Links (hard links, symbolic links) | |
| **UNIT-III:** | **10hrs** |
| Shell introduction and Shell Scripting: Shell and various type of shell, Various editors present in Unix, Different modes of operation in vi editor, Shell script, Writing and executing the shell script, Shell variable (user defined and system variables), System calls, Using system calls, Pipes and Filters. | |
| **UNIT-IV:** | **10hrs** |
| Unix Control Structures and Utilities: Decision making in Shell Scripts (If else, switch), oops in shell, Functions, Utility programs (cut, paste, join, tr, uniq utilities), Pattern matching utility (grep). | |
| **Text Books:** | |
| 1. Sumitabha, Das, Unix Concepts And Applications, Tata McGraw-Hill Education, Fourth Edition,2017 | |
| **Reference Books:** | |
| 1. Nemeth Synder & Hein, Linux Administration Handbook, Pearson Education, Second Edition,2010 | |

| | COURSE OUTCOMES:<br>After completion of this course successfully, the students will be able to- |
|---|---|
| CO1 | Be able to understanding basic operating system fundamentals |
| CO2 | Know how an operating system can be used as a service |
| CO3 | Familiarity with linux programming concepts |
| CO4 | Have a foundation stone to understand operating systems working |

**DSE – 2 Practical: Unix Programming Lab (Credit: 2 )**
**Marks:25**

1. Write a shell script to check if the number entered at the command line is prime or not.
2. Write a shell script to modify "cal" command to display calendars of the specified months.
3. Write a shell script to modify "cal" command to display calendars of the specified range of months.
4. Write a shell script to accept a login name. If not a valid login name display message "Entered login name is invalid".
5. Write a shell script to display date in the mm/dd/yy format.
6. Write a shell script to display on the screen sorted output of "who" command along with the total number of users.
7. Write a shell script to display the multiplication table of any number.
8. Write a shell script to compare two files and if found equal asks the user to delete the duplicate file.
9. Write a shell script to find the sum of digits of a given number.
10. Write a shell script to merge the contents of three files, sort the contents and then display them page by page.
11. Write a shell script to find the LCD (least common divisor) of two numbers.
12. Write a shell script to perform the tasks of basic calculator.
13. Write a shell script to find the power of a given number.
14. Write a shell script to find the greatest number among the three numbers.
15. Write a shell script to find the factorial of a given number.
16. Write a shell script to check whether the number is Armstrong or not.

## Semester-VI

## DSE-3: DATA SCIENCE

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

### COURSE OVERVIEW:

This course provides an overview of types of data and their importance. It includes the technologies to analyse the data and get meaningful insights. It helps the learners in applying different tools to investigate data for extracting decisive information. The course makes the students learn to visualise the data using various graphical aids that clarifies the understanding about the data.

### COURSE OBJECTIVES:

1. To learn emerging technologies related to data science.

2. To understand the underlying principles of data science, exploring data analysis.

3. To learn the basics of R Programming.

| **DSE-3** |
|---|
| **Data Science** |
| **UNIT-I:** **10hrs** |
| Data Scientist's Tool Box: Turning data into actionable knowledge, introduction to the tools that will be used in building data analysis software: version control, markdown, git, GitHub, R, and RStudio. |
| **UNIT-II:** **10hrs** |
| R Programming Basics: Overview of R, R data types and objects, reading and writing data, Control structures, functions, scoping rules, dates and times, Loop functions, debugging tools, Simulation, code profiling. |
| **UNIT-III:** **10hrs** |
| Getting and Cleaning Data: Obtaining data from the web, from APIs, from databases and from colleagues in various formats, basics of data cleaning and making data "tidy". |
| **UNIT-IV:** **10hrs** |
| Exploratory Data Analysis: Essential exploratory techniques for summarizing data, applied before formal modeling commences, eliminating or sharpening potential hypotheses about the world that can be addressed by the data, common multivariate statistical techniques used to visualize high-dimensional data. |
| **Text Books:** |
| 1.Rachel Schutt, Cathy O'Neil, "Doing Data Science: Straight Talk from the Frontiline", chroff/O'Reilly, First Edition, 2013. |
| **Reference Books:** |
| 1. Foster Provost, Tom Fawcett, "Data Science for Business What You Need to Know About Data Mining and Data-Analytic Thinking", O'Reilly, First Edition, 2013.<br>2. John W. Foreman, "Data Smart: Using data Science to Transform Information into Insight", John Wiley & Sons, First Edition, 2013.<br>3. Eric Seigel, "Predictive Analytics: The Power to Predict who Will Click, Buy, Lie, or Die", Wiley, Second Edition , 2016. |

| | **COURSE OUTCOMES:**<br>After completion of this course successfully, the students will be able to- |
|---|---|
| **CO1** | Understand the programming elements to develop code in R. |
| **CO2** | Learn to develop cost effective solutions to complex problems using appropriate built in functions. |
| **CO3** | Identify the objectives of target audience and explore various visualisation tools to explain the information. |
| **CO4** | Apply data science concepts and methods to solve problems in real-world contexts and will communicate these solutions effectively. |

**DSE-3 Practical: Elementary Data Science Lab (Credits: 2)**
**Marks:25**

1. Write a program that prints "Hello World" to the screen.
2. Write a program that asks the user for a number n and prints the sum of the numbers 1 to n
3. Write a program that prints a multiplication table for numbers up to 12.
4. Write a function that returns the largest element in a list.
5. Write a function that computes the running total of a list.
6. Write a function that tests whether a string is a palindrome.
7. Implement linear search.
8. Implement binary search.
9. Implement matrices addition, subtraction and Multiplication
10. Fifteen students were enrolled in a course. There ages were:
20 20 20 20 20 21 21 21 22 22 22 22 23 23 23
i. Find the median age of all students less than 22 years.
ii. Find the median age of all students.
iii. Find the mean age of all students.
iv. Find the modal age for all students.
v. Two more students enter the class. The age of both students is 23. What is now mean, mode and median?

## DSE-4: DATA MINING

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

### COURSE OVERVIEW:

This course provides an overview data-oriented technique for extracting patterns from data. It includes the basics of data warehouse and online analytical processing. It covers the techniques behind data pre-processing. The course makes the students learn to data mining tasks like association and classification.

### COURSE OBJECTIVES:

1. To introduce the basic concepts of data warehousing, data mining, issues, and implication.
2. To learn the core topics like association rules, classification, prediction and clustering techniques.
3. To make a study on the applications and trends in data mining.

| **DSE-4** | |
|---|---|
| **Data Mining** | |
| **UNIT-I:** | **10hrs** |
| Data Warehouse Fundamentals: Introduction to Data Warehouse, OLTP Systems, OLAP, Differences between OLTP and OLAP, Characteristics of Data Warehouse, Functionality of Data Warehouse, Advantages and Applications of Data Warehouse, Advantages, Applications, Top- Down and Bottom-Up Development Methodology, Tools for Data warehouse development, Data Warehouse Types, Data cubes. | |

| **UNIT-II:** | **10hrs** |
|---|---|
| Introduction to Data Mining: Data mining, Functionalities, Data Pre-processing: Pre-processing the Data, Data cleaning, Data Integration and Transformation, Data reduction, Discretization and Concept hierarchies. | |
| **UNIT-III:** | **10hrs** |
| Mining Association Rules: Basics Concepts – Single Dimensional Boolean Association Rules from Transaction Databases, Multilevel Association Rules from transaction databases, Multi dimension Association Rules from Relational Database and Data Warehouses. Apriori Algorithm, FP-Tree algorithm. | |
| **UNIT-IV:** | **10hrs** |
| Classification and Prediction: Introduction, Issues, Decision Tree Induction, Naïve Bayesian Classification, Classification based on Concepts from Association Rule Mining, Classifier Accuracy. | |
| **Text Books:** | |
| 1. J. Han and M. Kamber, "Data Mining Concepts and Techniques", Elsevier, Second Edition, 2011. | |
| **Reference Books:** | |
| 1. K.P. Soman ,Shyam Diwakar, V.Ajay ,"Insight into Data Mining Theory and Practice", Prentice Hall of India Pvt. Ltd, First Edition, 2010.<br>2. Arun K. Pujari, "Data Mining Techniques", Universities Press, Third Edition, 2013.<br>3. M. Panda, S. Dehuri, M. R. Patra, "Modern Approaches of Data Mining: Theory & Practice", Narosa Publishing House, First Edition, 2016. | |

| | **COURSE OUTCOMES:**<br>After completion of this course successfully, the students will be able to- |
|---|---|
| **CO1** | Understand the concepts of data warehousing. |
| **CO2** | Learn standard data mining methods and techniques such as association rules, data clustering and classification. |
| **CO3** | Identify the need of data transformation, analysis and pre-processing. |
| **CO4** | Apply data mining techniques on datasets of realistic sizes effectively using modern data analysis frameworks. |

**DSE-4 Practical: Data Mining Lab (Credits: 2)**
**Marks: 25**
**(Using Scilab/ MATLAB/ C/ Python/ R)**

1. Build a Data Warehouse and perform its operations.
2. Perform data pre-processing tasks and Demonstrate performing association rule mining on data sets.
3. Demonstrate performing classification on data sets.
4. Demonstrate performing clustering on data sets.
5. Demonstrate performing Regression on data sets.
6. Credit Risk Assessment. Sample Programs using German Credit Data.
7. Sample Programs using Hospital Management System.

# Skill Enhancement Course

**Semester-IV**

## SEC-2: PYTHON PROGRAMMING

**Credits: 4**
**Full Mark 100(End Term 80+Internal-20)**

**COURSE OVERVIEW:**

Python is generally use for Web Development,Artificial Intelligence,Machine Learning,Data Science, Data Visualization and Data Analysis.

**COURSE OBJECTIVES:**

1. To be able to know the basic concept of Python Language.
2. To use the tools to develop simple python programme.
3. To be able to design a program using different programming methodology.

| **Skill Enhancement Course-1** | |
|---|---|
| **Python Programming** | |
| **UNIT-I:** | **10 hrs** |
| **Planning the Computer Program**: Concept of problem solving, Problem definition,Program design, Debugging, Types of errors in Programming, Documentation. | |
| **UNIT-II:** | **10 hrs** |
| **Techniques of problem Solving**: Flowchart, decision table, algorithms, Structured programming concepts, Programming methodologies viz. top-down and bottom-up programming | |
| **UNIT-III:** | **10hrs** |
| **Overview of Programming:**Structure of a Python Program, Elements of Python, Introduction to Python: Python Interpreter, Using Python as calculator, Python shell, Indentation. Atoms, Identifiers and keywords, Literals, Strings, Operators (Arithmetic operator, Relational operator, Logical or Boolean operator, Assignment, Operator, Ternary operator, Bit wise operator,Increment and Decrement operator. | |
| **UNIT-IV:** | **10hrs** |
| **Creating Python Programs**: Input and Output Statements, Control statements (Branching, Looping, Conditional Statement, Exit function, Difference between break, continue and pass.), Defining Functions, default arguments. | |
| **Text Book:** | |
| 1. T. Budd, "Exploring Python", TMH, First Edition, 2011. | |
| **Reference Book:** | |
| 1. Allen Downey, Jeffrey Elkner, Chris Meyers , "How to think like a computer scientist : learning with Python" , Freely available online.2012 | |

**Online References:**

1.Python Tutorial/Documentation www.python.or 2015
2.http://docs.python.org/3/tutorial/index.html
3.http://interactivepython.org/courselib/static/pythonds
4.http://www.ibiblio.org/g2swap/byteofpython/read/

| | COURSE OUTCOMES:<br>After completion of this course successfully, the students will be able to- |
|---|---|
| CO1 | Understand the basic syntax and control statements. |
| CO2 | Able to create different python program using List,Tuple,Set and Dictionary.. |
| CO3 | Ability to understand the different methodology to design a software. |
| CO4 | Able to understand the different types of errors occur while developing a program |

# General Electives

## Semester-I

## GE-1: COMPUTER FUNDAMENTALS

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

### COURSE OVERVIEW:

Computer Fundamentals course is designed to understand the students about the different components used in the computer like I/O devices,memory.Explore them about number systems and about the latest technology like bluetooth,Big data and Cloud computing.

### COURSE OBJECTIVES:

1. To be able to know the basics of Computer.
2. To make them familier with different components of computer and its uses.
3. To learn the features of latest technology.

| General Elective-I | |
|---|---|
| **Computer Fundamentals** | |
| **UNIT-I:** | **10hrs** |
| **Introduction:** Introduction to computer system, uses, types. **Data Representation**: Number systems and character representation, binary arithmetic **Human Computer Interface**: Types of software, Operating system as user interface, utility programs | |
| **UNIT-II:** | **10hrs** |
| **Devices:** Input and output devices (with connections and practical demo), keyboard, mouse, joystick, scanner, OCR, OMR, bar code reader, web camera, monitor, printer, plotter **Memory:** Primary, secondary, auxiliary memory, RAM, ROM, cache memory, hard disks, optical disks. | |

| UNIT-III: | 10hrs |
|---|---|
| **Computer Organization and Architecture:** C.P.U., registers, system bus, main memory unit, cache memory, Inside a computer, SMPS, Motherboard, Ports and Interfaces, expansion cards, ribbon cables, memory chips, processors. | |
| **UNIT-IV:** | **10hrs** |
| **Overview of Emerging Technologies:** Bluetooth, cloud computing, big data, data mining, mobile computing and embedded systems. | |
| **Text Book:** | |
| 1. A. Goel, Computer Fundamentals, Pearson Education,First Edition 2010. | |
| **Reference Book:** | |
| 1.P. Aksoy, L. DeNardis, Introduction to Information Technology, Cengage Learning, 2009<br>2.P. K.Sinha, P. Sinha, Fundamentals of Computers, BPB Publishers, 2007 | |

| | **COURSE OUTCOMES:**<br>After completion of this course successfully, the students will be able to- |
|---|---|
| **CO1** | Understand the basic terms of computer systems |
| **CO2** | Able to know the different types of number systems and their conversion. |
| **CO3** | Ability to create different types documentations and presentation using MS Office. |
| **CO4** | Able to understand the basics of latest technology. |

**GENERIC ELECTIVE-1 Practical: Computer Fundamentals Lab (Credits: 2)**
**Marks:25**

Practical exercises based on MS Office tools including document preparation and spreadsheet handling packages.

**MS Word:**

1. Prepare a grocery list having four columns (Serial number, The name of the product, quantity and price) for the month of February, 2019. • Font specifications for Title (Grocery List): 14-point Arial font in bold and italics. • The headings of the columns should be in 12-point and bold. • The rest of the document should be in 10-point Times New Roman. • Leave a gap of 12-points after the title.

3. Create a telephone directory. • The heading should be 16-point Arial Font in bold • The rest of the document should use 10-point font size • Other headings should use 10-point Courier New Font. • The footer should show the page number as well as the date last updated.
4. Design a time-table form for your college. • The first line should mention the name of the college in 16-point Arial Font and should be bold. • The second line should give the course name/teacher‘s name and the department in 14- point Arial.

4. Create the following documents: a) A newsletter with a headline and 2 columns in portrait orientation, including at least one image surrounded by text.
(b) Use a newsletter format to promote upcoming projects or events in your classroom or college.
5. Enter the following data into a table given below:

| Salesperson | Dolls | Trucks | Puzzles |
|---|---|---|---|
| Kennedy, Sally | 1327 | 1423 | 1193 |
| White,Peter | 1421 | 3863 | 2934 |
| Pillar, James | 5214 | 3247 | 5467 |
| York,George | 2190 | 1278 | 1928 |
| Banks,Jennifer | 1201 | 2528 | 1203 |
| Atwater,Kelly | 4098 | 3079 | 2067 |

Add a column Region (values: S, N, N, S, S, S) between the Salesperson and Dolls columns to the given table Sort your table data by Region and within Region by Salesperson in ascending order: In this exercise, you will add a new row to your table, place the word "Total" at the bottom of the Salesperson column, and sum the Dolls, Trucks, and Puzzles columns.

**MS Excel**

6. Given the following worksheet

| | A | B | C | D |
|---|---|---|---|---|
| 1 | **Roll No.** | **Name** | **Marks** | **Grade** |
| 2 | 1001 | Sachin | 99 | |
| 3 | 1002 | Sehwag | 65 | |
| 4 | 1003 | Rahul | 41 | |
| 5 | 1004 | Sourav | 89 | |
| 6 | 1005 | Har Bhajan | 56 | |

Calculate the grade of these students on the basis of following guidelines:

| **If Marks** | **Then Grade** |
|---|---|
| >= 80 | A+ |
| >= 60 and < 80 | A |
| >= 50 and < 60 | B |
| < 50 | F |

7. Given the following worksheet

| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | **Salesman** | | **Sales in(Rs.)** | | | | |
| 2 | **No.** | **Qtr1** | **Qtr2** | **Qtr3** | **Qtr4** | **Total** | **Commission** |
| 3 | $001 | 5000 | 8500 | 12000 | 9000 | | |

| | | | | | |
|---|---|---|---|---|---|
| 4 | $002 | 7000 | 4000 | 7500 | 11000 |
| 5 | $003 | 4000 | 9000 | 6500 | 8200 |
| 6 | $004 | 5500 | 6900 | 4500 | 10500 |
| 7 | $005 | 7400 | 8500 | 9200 | 8300 |
| 8 | $006 | 5300 | 7600 | 9800 | 6100 |

Calculate the commission earned by the salesmen on the basis of following Candidates:

| If Total Sales | Commission |
|---|---|
| < 20000 | 0% of sales |
| > 20000 and < 25000 | 4% of sales |
| > 25000 and < 30000 | 5.5% of sales |
| > 30000 and <35000 | 8% of sales |
| >=35000 | 11% of sales |

The total sales is sum of sales of all the four quarters.

8. Create Payment Table for a fixed Principal amount, variable rate of interests and time in the format below:

| No. of Instalments | 5% | 6% | 7% | 8% | 9% |
|---|---|---|---|---|---|
| 3 | XX | XX | XX | XX | XX |
| 4 | XX | XX | XX | XX | XX |
| 5 | XX | XX | XX | XX | XX |
| 6 | XX | XX | XX | XX | XX |

9. A company XYZ Ltd. pays a monthly salary to its employees which consists of basic salary, allowances & deductions. The details of allowances and deductions are as follows:

**Allowances**

- HRA Dependent on Basic

30% of Basic if Basic <= 1000
25% of Basic if Basic > 1000 and Basic <= 3000
20% of Basic if Basic > 3000

- DA Fixed for all employees. 30% of Basic

- Conveyance Allowance

Rs 50/- if Basic is <=1000
Rs 75/- if Basic is >1000 and Basic <=2000
Rs 100/- if Basic is >2000

**Deductions**

- Provident Fund 6% of Basic

- Group Insurance Premium Rs. 40/- if Basic is <= 1500
Rs. 60/- if Basic is > 1500 and Basic <= 3000
Rs. 80/- if Basic is > 3000

Calculate the following:

Gross Salary = Basic + HRA + DA + Conveyance + Entertainment

Total Deduction=Provident Fund + Group Insurance Premium

Net Salary=Gross Salary - Total deduction

9. The following table gives year wise sale figure of five salesman in Rs.

| **Salesman** | **2000** | **2001** | **2002** | **2003** |
|---|---|---|---|---|
| s1 | 10000 | 12000 | 20000 | 50000 |
| **s2** | 15000 | 18000 | 50000 | 60000 |
| S3 | 20000 | 22000 | 70000 | 70000 |
| s4 | 30000 | 30000 | 100000 | 80000 |
| s5 | 40000 | 45000 | 125000 | 90000 |

(a) Calculate total sale year wise.
(b) Calculate the net sale made by each salesman.
(c) Calculate the maximum sale made by the salesman.
(d) Calculate the commission for each salesman under the condition.
(i) If total sales >400000 give 5% commission on total sale made by the salesman.
(ii) Otherwise give 2% commission.
(e) Draw a bar graph representing the sale made by each salesman
(f) Draw a pie graph representing the sale made by salesman in 2000.

**Semester-II**

# GE-2: C AND DATA STRUCTURE

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

## COURSE OVERVIEW:

C programming language is an important language for a student to develop himself in a programming field. Data structure is an important concept which are used in various fields of computer like Database, AI, Machne Learning,Numerical data Analysis

.

## COURSE OBJECTIVES:

1. To be able to know the basics of Programming language.
2. To be able to write different programs and understand the use of data structure in a program..
3. To learn the sorting and searching of data in linear Data Structure

| **Generic Elective-2** | |
|---|---|
| **C and Data Structure** | |
| **UNIT-I:** | **10hrs** |
| Algorithm, flowchart, program development steps, structure of C program, A Simple C program, identifiers, basic data types and sizes, Constants, variables, arithmetic, relational and logical operators, increment and decrement operators, conditional operator, bit-wise operators,assignment operators, expressions, type conversions, conditional expressions, precedence and order of evaluation. Input-output statements, statements and blocks, if and switch statements, loops- while, do-while and for statements, break, continue, goto and labels, programming examples. | |
| **UNIT-II:** | **10hrs** |
| Designing structured programs, Functions, basics, parameter passing, storage classes- extern, auto, register, static, scope rules, block structure, user defined functions, standard library functions, recursive functions, header files, C preprocessor, example c programs. | |
| **UNIT-III:** | **10hrs** |
| Arrays- concepts, declaration, definition, accessing elements, storing elements, arrays and functions, two-dimensional and multi-dimensional arrays, applications of arrays. pointers concepts, initialization of pointer variables, pointers and function arguments, address arithmetic, Character pointers and functions, pointers to pointers, pointers and multidimensional arrays, dynamic memory managements functions, command line arguments, C program examples. | |
| **UNIT-IV:** | **10hrs** |
| Introduction to data structures, representing stacks and queues in C using arrays, infix to post fix conversion, postfix expression evaluation, Applications of Queue. Searching - Linear and binary search methods, sorting - Bubble sort, selection sort, Insertion sort, Quick sort. | |
| **Text Books:** | |

| |
|---|
| 1.E. Balagurusamy, “Programming in ANSI C”, Eighth Edition,Tata McHill, 2019<br>2. Seymour Lipschutz, “Data Structure with C”, Schaum’s Outlines MGH.,2017 |
| **Reference Book:** |
| 1. B. Kernighan & Dennis Ritchie, “The C Programming Language”, Second Edition, Pearson,2015.<br>2. P.C. Sethi, P.K. Behera, “Programming using C”, Kalyani Publisher, Ludhiana ,2014<br>3. A.S.Tanenbaum, Y.Langsam, M.J. Augenstein, “Data Structures Using C”, PHI/Pearson,Seventh Edition, 2009 |

| | **COURSE OUTCOMES:**<br>After completion of this course successfully, the students will be able to- |
|---|---|
| **CO1** | Understand the basics of programming language and Data Structure. |
| **CO2** | Able to write programs using c and Data Stricture. |
| **CO3** | Understand the concept of Linear Data Structure. |
| **CO4** | Able to develop different types of console application. |

**GENERIC ELECTIVE -2 Practical: C and Data Structure Lab (Credits: 2)**
**Marks:25**

1. Write a Program to find the greatest among three numbers.
2. Write a Program to check a number is leap year or not.
3. Write a Program to print the sum and product of digits of an integer.
4. Write a Program to reverse a number.
5. Write a Program to compute the sum of the first n terms of the following series
   S = 1+1/2+1/3+1/4+……
6. Write a function to find whether a given no. is prime or not.
7. Write a Program to compute factorial of a number.
8. Write a Program to print a triangle of stars as follows (take number of lines from user):

```
*
***
*****
*******
```

9. Write a program which takes the radius of a circle as input from the user, passes it to another function that computes the area and the circumference of the circle and displays the value of area and circumference from the main( ) function.
10. To insert and delete elements from appropriate position in an array.
11. To search an element and print the total time of occurrence in the array.
12. Array implementation of Stack.
13. Array implementation of Queue.
14. To perform Bubble sort.
15. To perform Selection sort.

**Semester-III**

# GE-3: PYTHON PROGRAMMING

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

## COURSE OVERVIEW:

Python programming language is basically used for web development..It playi an important role in the field of Data Science like AI, Machine Learning, Data Analysis and Data visualization.

.

## COURSE OBJECTIVES:

1. To enable the students to understand the core principles of the Python Language. 
2. To use the tools to produce well designed programs in python. 
3. To create effective GUI applications.

| **Generic Elective-3** | |
|---|---|
| **Python Programming** | |
| **UNIT-I:** | **08hrs** |
| Introduction to Python: Python Interpreter, Python as calculator, Python shell, Indentation, identifier and keywords, literals, strings, operatory (Arithmetic, Relational or decrement operator). Input output statement, control statements, (Branding, looping, conditional statement, Exit function) | |
| **UNIT-II:** | **10hrs** |
| String manipulations: Subscript operator, indexing, slicing a string, other functions on strings string module. Strings and number system, format functions: converting strings to numbers & Vice Versa. List, tuples, sets, Dictionaries: Basic list operators, replacing, inserting, removing an element, searching, Sorting lists, dictionary literals, adding & removing keys, accessing & replacing values, traversing dictionaries , Array in Python. | |
| **UNIT-III:** | **12hrs** |
| Design with Functions: hiding redundancy, complexity, arguments & return values; Formal/Actual arguments, named arguments, program structure and design, Recursive functions, scope & Global statements, Importing modules, Math modules & Random modules. Exception Handling: Exceptions, except clause, try and finally clause, user defined exceptions. File Handling: Manipulating files & directories, OS & SYS modules, Reading, Writing text & numbers from/to file. | |
| **UNIT-IV:** | **10hrs** |
| Simple Graphics: “Turtle” module; simple drawing colors, shapes, digital images, image file formats, Graphical U&S interfaces: Event driver programming, Paradigm, tkinter module, creating. Simple GUI: buttons, labels entry fields, dialogs, widget attributes-sizes fonts, colors, layout. | |
| **Text Book:** | |
| 1. Reema Thareja,”Python Programming using problem solving approach”, Oxford University Press.,2nd Edition 2023. | |
| **Reference Books:** | |
| 1. John V.Guttag” Introduction to Computation and Programming Using Python With Application to | |

| |
|---|
| Computational Modeling and Understanding Data",MIT Press, 2021<br>2.Charles Diorbach,,"Introduction to Computer Science using Python",Wiley,1st Edition,2015 |

| | **COURSE OUTCOMES:**<br>After completion of this course successfully, the students will be able to- |
|---|---|
| **CO1** | Understand the syntax and efficient in core python programming. |
| **CO2** | Proficiency in the handling of string and functions of python. |
| **CO3** | Able to understand the different modules, file systems. And exception handling. |
| **CO4** | Able to develop Graphics program using object oriented concept. |

**GENERIC ELECTIVE-3Practical: Programming in Python Lab (Credits: 2)**
**Marks:25**

1. Using for loop, print a table of Celsius/Fahrenheit equivalences. Let c be the Celsius temperatures ranging from 0 to 100, for each value of c, print the corresponding Fahrenheit temperature.
2. Using while loop, produce a table of sins, cosines and tangents. Make a variable x in range from 0 to 10 in steps of 0.2. For each value of x, print the value of sin(x), cos(x) and tan(x).
3. Write a program that reads an integer value and prints —leap year or —not a leap year .
4. Write a program that takes a positive integer n and then produces n lines of output shown as follows. For example enter a size: 5

```
*
**
***
****
*****
```

5. Write a function that takes an integer _n' as input and calculates the value of 1 + 1/1! + 1/2! + 1/3! + … + 1/n
6. Write a function that takes an integer input and calculates the factorial of that number.
7. Write a function that takes a string input and checks if it's a palindrome or not.
8. Write a list function to convert a string into a list, as in list (_abc') gives [a, b, c].
9. Write a program to generate Fibonacci series.
10. Write a program to check whether the input number is even or odd.
11. Write a program to compare three numbers and print the largest one.
12. Write a program to print factors of a given number.
13. Write a method to calculate GCD of two numbers.
14. Write a program to create Stack Class and implement all its methods. (Use Lists).
15. Write a program to create Queue Class and implement all its methods. (Use Lists)
16. Write a program to implement linear and binary search on lists.
17. Write a program to sort a list using insertion sort and bubble sort and selection sort

## Semester-IV

# GE-4: WEB TECHNOLOGY

**Credits: 4**
**Full Mark 75(End Term 60+Internal-15)**

### COURSE OVERVIEW:

Web technologies are concerned with how computers communicate over the web.How the systems connect and interact with the web by using the languages of the internet(HTML, CSS, JS, PHP) and allow to create different types of web sites to be shared worldwide.

.

### COURSE OBJECTIVES:

1. To learn the fundamentals of web designing. 
2. To design and develop standard and interactive web pages. 
3. To learn some popular web scripting languages.

| **Generic Elective-4** |
|---|
| **Web Technology** |
| **UNIT-I:** **08hrs** |
| Web Essentials: Clients, Servers and Communication: The Internet – Basic Internet protocols – The WWW, HTTP request message – response message, web clients web servers – case study. Introduction to HTML: HTML, HTML domains, basic structure of an HTML document – creating an HTML document, mark up tags, heading, paragraphs, line breaks, HTML tags. Elements of HTML, working with text, lists, tables and frames, working with hyperlink, images and multimedia, forms and controls |
| **UNIT-II:** **10hrs** |
| Introduction to cascading style sheets: Concepts of CSS, creating style sheet, CSS properties, CSS styling (background, text format, controlling fonts), working with the block elements and objects. Working who lists and tables, CSS ID and class. Box model (introduction, border properties, padding properties, margin properties), CSS colour, groping, Dimensions, display, positioning, floating, align, pseudo class, Navigation bar, image sprites. |
| **UNIT-III:** **12hrs** |
| Java scripts: Client side scripting, what is java script, simple java script, variables, functions, conditions, loops and repetitions. Java scripts and objects, java script own objects, the DOM and web browser environment, forms and validations. DHTML: Combining HTML, CSS, java scripts, events and buttons, controlling your browser |
| **UNIT-IV:** **10hrs** |
| PHP: Starting to script on server side, PHP basics, variables, data types, operators, expressions, constants, decisions and loop making decisions. Strings – creating, accessing strings, searching, replacing and formatting strings. Arrays: Creation, accessing array, multidimensional arrays, PHP with Database. |
| **Text Book:** |
| 1. Kogent Learning Solutions, “Web Technologies Black Book”,Dreamtech,1st Edition,2009. |

| |
|---|
| 2. Matt.Doyle,"Beginning PHP 5.3",Wiley Publishing,1st Edition,2011.<br>3. John Duckett,"Beginning with HTML, XHTML, CSS and Java Script",Indian Edition, 2009 |
| **Reference Book:** |
| 1. Steven M. Schafer, "HTML, XHTML and CSS Bible" , Willey India, 5th Edition, 2010. |

| | **COURSE OUTCOMES:**<br>After completion of this course successfully, the students will be able to- |
|---|---|
| **CO1** | Create dynamic web pages using css and js language. |
| **CO2** | Able to connect a programme like java, php to a DBMS. |
| **CO3** | Able to understand the client side and server side scripting language. |
| **CO4** | Able to develop a server side application fetching the data from clients and store the data in database. |

**GENERIC ELECTIVE-4 Practical: Web Technology Lab (Credits: 2)**
**Marks:25**

1. Practicing basic and advanced text for formatting.
2. Practice use of image, video and sound in HTML documents.
3. Designing of web pages- Document layout, list, tables.
4. Practicing Hyperlink of web pages, working with frames.
5. Working with forms and controls.
6. Acquaintance with creating style sheet, CSS properties and styling.
7. Working with background, text, font, list properties.
8. Working with HTML elements box properties in CSS.
9. Develop simple calculator for addition, subtraction, multiplication and division operation using java script.
10. Create HTML page with java script which takes integer number as a input and tells whether the number is odd or even.
11. Create HTML page that contains form with fields name, Email, mobile number, gender, favorite colour and button; now write a java script code to validate each entry. Also write a code to combine and display the information in text box when button is clicked.
12. Write a PHP program to check if number is prime or not.
13. Write a PHP program to print first ten Fibonacci numbers.
14. Create a MySQL data base and connect with PHP.
15. Write PHP script for string and retrieving user information from my SQL table. (a) Write a HTML page which takes Name, Address, Email and Mobile number from user (register PHP). (b) Store this data in MySQL data base. (c) Next page display all user in HTML table using PHP (display PHP).
16. Using HTML, CSS, Javascript, PHP, MySQL, design a authentication module of a web page

## Certificate Course

# VA-1: MACHINE LEARNING USING PYTHON

**Credits:**
**Full Mark:**

### COURSE OVERVIEW:

This course introduces the fundamental concepts behind supervised, unsupervised & reinforcement learning, assess & select appropriate model and use cross validation to tune their parameters. Most of the methods learned in this course are implemented using Python.

### COURSE OBJECTIVES:

4. To learn various machine learning methods.
5. To be able to implement different machine learning methods using Python programming language.
6. To be able to solve machine learnong related problems.

| VA-1 | |
|---|---|
| **Machine Learning using Python** | |
| **UNIT-I:** | **10hrs** |
| Introduction to Machine Learning, Supervised and Unsupervised Machine Learning. Introduction to Regression, Simple Linear Regression, Evaluation Metrics in Regression Models, Multiple Linear Regression. | |
| **UNIT-II:** | **10hrs** |
| Classification, K-Nearest Neighbours, Evaluation Metrics in Classification, Decision Trees, Building Decision Tree, Intro to Logistic Regression, Logistic Regression Training, Support Vector Machine (SVM), Kernel Function, SVM for the linearly non-separable dataset, SVM for more than two class problems. | |
| **UNIT-III:** | **10hrs** |
| Neural Network- McP Neuron, Perceptron Learning, Multi-layer perceptron, Backpropagation Learning. | |
| **UNIT-IV:** | **10hrs** |
| Clustering, Issues in clustering, k-Means algorithm, K selection in K-Means, K--medoid, Hierarchical Clustering | |
| **Text Book:** | |
| 1. T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning - Data Mining, Inference, and Prediction, Springer, 2nd Edition, 2009.<br>2. S. Haykin, Neural Networks and Learning Machines, Pearson Education, 3rd Edition, 2009.<br>3. E. Alpaydın, Introduction to Machine Learning, Prentice Hall of India, 2nd Edition, 2010. | |
| **Reference Books:** | |
| 1. Y. G. James, D. Witten, T. Hastie, and R. Tibshirani, An Introduction to Statistical Learning with Applications in R, Springer, 2nd Edition, 2013.<br>2. T. M. Mitchell, Machine Learning, McGrow-Hill Education, 1st Edition, 2013.<br>3. C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 1st Edition, 2006. | |

| | COURSE OUTCOMES:<br>After completion of this course successfully, the students will be able to- |
|---|---|
| CO1 | Apply the concepts of supervised machine learning and its functionalities. |
| CO2 | Determine most appropriate model in a specific context using model selection techniques. |
| CO3 | Perform classification using Bayes classifier, SVM, Decision Tree, and Random Forest. |
| CO4 | Reduce dimensionality using feature selection and apply unsupervised machine learning for solving problems. |

## VA-1 Practical: Machine Learning using Python

1. Implement simple linear regression using Python and test it with an example scenario.
2. Implement multiple linear regression using Python and test it with an example scenario.
3. Implement k-nn classification algorithm using Python and test it with an example scenario.
4. Implement Decision Tree using Python and test it with an example scenario.
5. Implement SVM using Python and test it with an example scenario.
6. Implement SVM with kernel using Python and test it with an example scenario.
7. Implement perceptron using Python and test it with an example scenario.
8. Implement backpropagation algorithm using Python and test it with an example scenario.
9. Implement k-means clustering using Python and test it with an example scenario.
10. Implement k-medoid clustering technique using Python and test it with an example scenario.