

DEPARTMENT OF I.T & COMPUTER SCIENCE

COURSE OUTCOMES

SEMESTER-I

C:1- (Programming Using C)

UNIT1-

This unit covers the C Programming language starting with the structure, constants and variable declarations, the main subroutine, simple input/output, arithmetic expressions, Boolean expressions, the assignment statement, and lastly simple functions with at most one input and one output.

Unit2-

In programming the order of execution of instructions may have to be changed depending on certain conditions. This involves a kind of decision making to see whether a particular condition has occurred or not and then direct the computer to execute certain instructions accordingly.

Unit3-

A function is a block of code that performs a particular task. There are some situations when we need to write a particular block of code for more than once in our program. C language provides an approach in which you need to declare and define a group of statements once and that can be called and used whenever required. This saves both time and space.

Unit4-

Yes pointers are variable but variable that can point to the address of other variable . One use of pointer in C is pass-by-reference that when you pass a value lets say from main() to a function it is by default passed by value that mean if the function does any change to that variable that is passed it won't affect the one in main() but using pass-by-reference it will change the value in main().

Unit5-

A **file** represents a sequence of bytes on the disk where a group of related data is stored. File is created for permanent storage of data. It is a readymade structure. In C language, we use a structure **pointer of file type** to declare a file.

C:2-(Computer Organization)

Unit1-After going through this unit, you will be able to understand the decimal, binary, octal and hexadecimal number systems, convert from one number system into another, apply arithmetic operations to binary numbers, understand BCD codes and alpha numeric codes, learn the operations of logic gates, apply the basic laws of Boolean algebra, apply De Morgan's theorems to Boolean expressions

Unit 2-

In electronics, a flip-flop or latch is a circuit that has two stable states and can be used to store state information. A flip-flop is a bitable multivibrator.

These Registers are used for performing the various Operations. While we are working on the System then these Registers are used by the CPU for Performing the Operations.

Unit 3-

In computer architecture, a bus is a communication system that transfers data between components inside a computer, or between computers. Each instruction initiates a sequence of micro operations that fetch operands from registers or memory, possibly perform arithmetic, logic, or shift operations, and store results in registers or memory. Instructions are encoded as binary instruction codes.

Unit 4-

In computer programming, a subroutine is a sequence of program instructions that perform a specific task, packaged as a unit. This unit can then be used in programs wherever that particular task should be performed. A stack is a container of objects that are inserted and removed according to the last-in first-out (LIFO) principle. In the pushdown stacks only two operations are allowed: push the item into the stack, and pop the item out of the stack. A queue is a container of objects (a linear collection) that are inserted and removed according to the first-in first-out (FIFO) principle.

Unit 5-

Understand the concepts of interconnecting processor to memory devices, understand the speed of access of memory-devices, latency and bandwidth, and learn how they relate in a memory system

SEMESTER-II

C:3-(Programming Using C++)

Unit1-

Object-oriented programming (OOP) is a programming paradigm based on the concept of "objects", which may contain data, in the form of fields, often known as *attributes* and code, in the form of procedures, often known as methods. A feature of objects is that an object's procedures can access and often modify the data fields of the object with which they are associated. In OOP, computer programs are designed by making them out of objects that interact with one another. The most popular ones are class-based, meaning that objects are instances of classes, which typically also determine their type.

Unit2-

In C++, a function is a group of statements that is given a name, and which can be called from some point of the program. The most common syntax is: type name (parameter1, parameter2,) {statements}. A class is an abstract data type similar to 'C structure'. Class representation of objects and the sets of operations that can be applied to such objects. Class consists of Data members and methods. An **Object** is an instance of a Class. When a class is defined, no memory is allocated but when it is instantiated (i.e. an object is created) memory is allocated.

Unit3-

Constructors are special class functions which performs initialization of every object. The Compiler calls the Constructor whenever an object is created. Constructors' initialize values to object members after storage are allocated to the object. Destructor is a special class function which destroys the object as soon as the scope of object ends. The destructor is called automatically by the compiler when the object goes out of scope.

Operator overloading means that the operation performed by the operator depends on the *type* of operands provided to the operator. Conversion operators help to cast user-defined types from one to the other much like the basic types.

Unit4-

Inheritance is one of the key features of Object-oriented programming in C++. It allows user to create a new class (derived class) from an existing class (base class).The derived class inherits all the features from the base class and can have additional features of its own. The word polymorphism means having many forms. Typically, polymorphism occurs when there is a hierarchy of classes and they are related by inheritance. virtual functions are called according to the type of object pointed or referred, not according to the type of pointer or reference. In other words, virtual functions are resolved late, at runtime. To access address of a variable to a pointer, we use the unary operator **&** (ampersand) that returns the address of that variable.

Unit5-

In case of C++ it uses streams to perform input and output operations in standard input output devices (keyboard and monitor). A stream is an object which can either insert or extract the character from it. An implementation file is used in C++ programming when creating a class definition to split the interface from the implementation. The header file would declare all the member functions (methods) and data methods (fields).

C:4-(Data Structure)

Unit1-

Data Structure is a way of collecting and organizing data in such a way that we can perform operations on these data in an effective way. Data Structures is about rendering data elements in terms of some relationship, for better organization and storage.

Unit2-

A **linked list** is a linear collection of data elements, in which linear order is not given by their physical placement in memory. Instead, each element points to the next. It is a data structure consisting of a group of nodes which together represent a sequence.

Unit3-

The simplest application of a stack is to reverse a word. You push a given word to stack - letter by letter - and then pop letters from the stack. Another application is an "undo" mechanism in text editors; this operation is accomplished by keeping all text changes in a stack.

Unit4-

A queue is a container of objects (a linear collection) that are inserted and removed according to the first-in first-out (FIFO) principle. An excellent example of a queue is a line of students in the food court of the UC. The difference between stacks and queues is in removing. In a stack we remove the item the most recently added; in a queue, we remove the item the least recently added.

Unit5-

A **tree** is a widely used abstract data type (ADT)—or data structure implementing this ADT—that simulates a hierarchical tree structure, with a root value and sub trees of children with a parent node, represented as a set of linked nodes.

SEMESTER-III

C:5 - (Operating system)

Unit1-

An Operating System (OS) acts as an interface connecting a computer user with the hardware of the computer. An operating system falls under the category of system software that performs all the fundamental tasks like file management, memory handling, process management, handling the input / output and governing and managing the peripheral devices like disk drives, networking hardware, printers etc.

Unit2-

Scheduling processes on the processor is often called "process scheduling" or simply "scheduling". Fairness, Efficiency, Low response time, Low turnaround time, High throughput, Repeatability.

Unit3-

CPU scheduling, the basis for multiprogrammed operating systems. CPU-scheduling algorithms. Evaluation criteria for selecting a CPU-scheduling algorithm for a particular system. Scheduling algorithms of several operating systems.

Unit4-

To develop a description of deadlocks, which prevent sets of concurrent processes from completing their tasks. To present a number of different methods for preventing or avoiding deadlocks in a computer system.

Unit5-

The computer's operating system, using a combination of hardware and software, maps memory addresses used by a program, called virtual addresses, into physical addresses in computer memory. Guarantees data in the file is valid, Optimizes performance in terms of throughput & response time, Provide I/O support for storage device type, Provide I/O support for multiple users, Meet user requirements for data operations.

C:6 - (Database Management System)

Unit1-

In the database, each set of information is stored in the form of rows and columns. We define a unique key column for each record known as primary key. This helps in reducing unnecessary data storage and faster retrieval of data. A **data model** is an abstract model that organizes elements of data and standardizes how they relate to one another and to properties of the real world entities.

Unit2-

relational algebra is a (high-level) procedural language; can tell the DBMS how to build a new relations from other relations in the database. Relational calculus is an query language which is non procedural, and instead of algebra it uses mathematical predicate calculus.

Unit3-

SQL (Structured Query Language) is a standardized programming language used for managing relational databases and performing various operations on the data in them. The uses of SQL include modifying database table and index structures; adding, updating and deleting rows of data; and retrieving subsets of information from within a database for transaction processing and analytics application.

Unit4-

Normalization is the procedure to split the relation into relations with less attributes thereby minimizing the redundancy of the data and minimizing the insertions, deletions and updating. There are above five normal forms. These are 1NF, 2NF, 3NF, 4NF and 5NF. Here NF stands Normal Form.

Unit5-

Hash File organization method is the one where data is stored at the data blocks whose address is generated by using hash function.

C:7-(Discrete structure)

Unit1-

Discrete mathematics deals with objects that come in discrete bundles. A predicate with variables can be made a proposition by either assigning a value to the variable or by quantifying the variable. Nested quantifier uses a quantifier that appears within the scope of another quantifier. Structural induction applies induction on recursive definitions even if there is no integer. Algorithm recursive solves by reducing it to an instance of the problem with smaller input.

Unit2-

Recursive function used to describe any function that is defined with recursion. Subsets of n -ary Cartesian Products. The Cartesian product is the set of n -tuples $(a_1 \dots a_n)$. Fundamental of equivalence relations, relation and properties, partial ordering, Boolean (true or false).

Unit3-

Boolean functions used for a description of the operation of discrete control systems. Logic gates are the basic building blocks of logic circuits. Study of structure of algebras, monodies, homomorphism and congruence relation. A polynomial **ring** is a set equipped with addition and multiplication satisfying certain properties.

Unit4-

Pigeonhole principle of n items is put into m containers. Concepts in fundamental counting rule, the *permutation* rule, and the *combination* rule. Binomial coefficient, generating perambulations and combinations. *Inclusion–exclusion principle* is to the combinatorial problem of counting all derangements of a finite set. **Bayes' theorem** that describes how to update the probabilities of hypotheses when given evidence.

Unit5-

Graph theory is an important part of a number of disciplines in the fields of mathematics, engineering and computer science. *Graph Models: Social Networks. Model social structures*: relationships between people or groups. *Havel-Hakim* method, determine if the degree sequence numbers in graphical. Study of graph isomorphism and connectivity. **Euler path** and **Hamiltonian path** graph problems deal with finding a path between two vertices or finding a path between two vertices. Fundamental of network flows and graph coloring.

SEMESTER-IV

C:8-(Java programming)

Unit1-

Understand fundamentals of programming such as variables, conditional and iterative execution, methods. Understand fundamentals of object-oriented programming in Java, including defining classes, data types, operator, invoking methods, using class libraries; be aware of the important topics and principles of software development. Be able to use the Java SDK environment to create, debug and run simple Java programs.

Unit 2-

An introduction to object oriented programming (OOP) using the Java programming language. The model of object oriented programming: string class, class, byte, character stream, class constructor, method overloading. Fundamental features of an object oriented language like Java: object classes and interfaces, exceptions and libraries of object collections.

Unit 3-

An introduction to purely (OOP) using the Java programming language. In this unit using abstract data types, encapsulation, inheritance and polymorphism. Using standard java packages like Lang, io, net can import individual classes or a whole package of classes.

Unit 4-

Multithreading in java is a process of executing multiple threads, difference between multithreading, multiprocessing; threads share a common memory area. An exception is an object that wraps an error event that occurred within a method and contains: Information about the error including its type. Study of Internet protocol and data programming. Database connectivity, manipulating, accessing with JDBC (java database connectivity).

Unit 5-

Applet program that is embedded in the webpage. How to runs inside the browser and works at client side. Event handling of click on button, dragging mouse use of java.awt.event package using many event classes and Listener interfaces for event handling. Design and implementation of GUIs using AWT controls. Application diagram and calculation showing through lines, rectangle and oval, Fetching of fonts. Textbox label buttons creation and implement.

C:9-(Computer Network)

Unit1-

The Open Systems Interconnection model (OSI model) is a conceptual model that characterizes and standardizes the communication functions of a telecommunication or computing system without regard to their underlying internal structure and technology. Its goal is the interoperability of diverse communication systems with standard protocols. The model partitions a communication system into abstraction layers. The original version of the model defined seven layers.

Unit2-

There is different type of signal conversion i.e. from analog to digital, digital to digital, digital to analog and analog to analog.

Unit3-

In telecommunications and computer networks, multiplexing is a method by which multiple analog or digital signals are combined into one signal over a shared medium. There are basically three types of switching methods are made available. Out of three methods, circuit switching and packet switching are commonly used but the message switching has been opposed out in the general communication procedure but is still used in the networking application.

Unit4-

error detection and correction or error control are techniques that enable reliable delivery of digital data over unreliable communication channels. Many communication channels are subject to channel noise, and thus errors may be introduced during transmission from the source to a receiver. Error detection techniques allow detecting such errors, while error correction enables reconstruction of the original data in many cases.

Unit5-

Stands for "Local Area Network" and is pronounced "lan." A LAN is a network of connected devices that exist within a specific location. LANs may be found in homes, offices, educational institution, or other areas. A LAN may be wired, wireless, or a combination of the two. A standard wired LAN uses Ethernet to connect devices together. Wireless LANs are typically created using a Wi-Fi signal. If a router supports both Ethernet and Wi-Fi connections, it can be used to create a LAN with both wired and wireless devices.

C:10-(Computer Graphics)

Unit-1

Graphic Systems Industrial & Commercial Printing & Packaging *monitor* or *display* is electronic visual *display* for computers. Raster scan system the electron beam is swept across the screen, one row at a time from top to bottom. Input devices, hard copy devices, graphic software. Graphic point and lines draw. *Algorithm* graphical *algorithm* for a *line* segment on discrete graphical media. Graphics primitive, no divisible graphical element for input or output within a computer-graphics system. File area attributes. Antialiasing software technique for diminishing lines.

Unit-2

Basic geometric transformations are translation, rotation, and scaling. Other transformations that are often applied to objects include reflection and shear. Study of matrix representation, homogenous coordinates, composite transformations, inverse transformation. Two Dimensional *Viewing. Pipeline*. Clipping window -- the part of two dimensional scenes that it to be displayed, viewport coordinate transformation. Line clipping is the process of removing lines or portions of lines outside an area of interest. 3D can view an object from any spatial position, Viewing Transformation. *Parallel Projection. Perspective Transformation* and Perspective Projection.

Unit-3

Curved Surface Area of a Cylinder. Quadratic surfaces include the cone, cylinder, ellipsoid, sphere, and spheroid. *B-spline*, or basis spline, is a spline function that has minimal support with respect to a given degree. *BSP tree* is a hierarchical subdivision of n dimensional space. Fractal geometry methods.

Unit-4

Visible-Surface Detection identifies those parts of a scene that are *visible* from a chosen viewing position. Back face detection and depth buffer method. *Scan line. Z – Sort. Depth List*. Visible element determination. Ray casting is the use of ray–surface intersection tests.

Unit-5

Illumination Model Reflection – light is reflected with equal intensity in all directions. Displaying light intensities, *Halftone screen patterns* are a key ingredient of retro design, polygon rendering methods. Ray tracing's in a realistic simulation of lighting over other rendering methods. *Computer animation* process used for generating *animated* images. Hierarchical modeling introductory.

SEC:II-(Android programming)

Unit-1

Android is a mobile operating system developed by Google, designed primarily for touch screen mobile devices such as smart phones and tablets. Android developer *tools* to build apps for *Android* phones, tablets. OOPS programming concept using java. An abstract computing machine to run a Java program.

Unit-2

A programming tool or software development tool is a computer program that software developers use to create, debug, and maintain. Eclipse development application with ADT plug in. Android emulator of sandwich jellybean, creating a small program. Deploying in android devices.

Unit-3

Functional requirements for a user interface system. *Lifecycle* callback methods. Device screen size deploying like height, width inches.

Unit-4

Form appearing with applies action, textbox use, button event, toggle button for different devices. Dropdown box, putting images, dialog box of messages, and stylish menu use.

Unit-5

Database connectivity with android application and uses of sqlite database query.

SEMESTER-V

C:11-(Internet Technology)

Unit1-

Objects are key to understanding object-oriented technology. Look around right now and you'll find many examples of real-world objects: The Array List class extends Abstract List and implements the List interface.

Unit2-

JavaScript is an excellent language to write object oriented web applications. It can support OOP because it supports inheritance through prototyping as well as properties and methods.

Unit3-

To connect with individual databases, JDBC (the Java Database Connectivity API) requires drivers for each database. The JDBC driver gives out the connection to the database and implements the protocol for transferring the query and result between client and database.

Unit4-

Java Server Pages (JSP) is a technology that helps software developers create dynamically generated web pages based on HTML, XML, or other document types. Released in 1999 by Sun Microsystems.

Unit5-

In computing based on the Java Platform; JavaBeans are classes that encapsulate many objects into a single object (the bean). They are serializable, have a zero-argument constructor, and allow access to properties using getter and setter methods. The name "Bean" was given to encompass this standard, which aims to create reusable software components for Java.

C:12-(Software Engineering)

Unit-1

Introduction to *Professional Software Development* work in teams, to find and fix errors, to write software and insures their code base against disaster. *Software engineering* leaders promote an *ethical* approach to the management of *software* development and maintenance. Software process model analysis, process activities and coping with change. Agile software development set of values and principles for software development under which requirements and solutions. Agile method, agile development agile project management and scaling agile methods

Unit-2

Requirements engineering refers to the process of defining, documenting and maintaining requirements in the engineering design process. Non-functional requirement is a requirement that specifies criteria can be used to judge the operation of a system contrasted with functional requirements. Software requirement documents, requirement specification, engineering process requirement analysis, requirement validation, requirement management, system modeling, context models, interaction models, structural models, behavior models. *Architectural design* news on *Architectural Digest*, including profiles on top talent, building announcements and new projects. *Architecture decision* record is a short text file. Architectural views and patterns.

Unit-3

Object-Oriented Analysis and *Design Concepts* in *UML*, inherently *object-oriented* modeling language designed for use in *object-oriented* software applications. *UML* and *OOAD*. *UML* core elements. *Open-source software development* is the process by *open-source* software, source code is publicly available. Software testing after development, test driven development, release testing, software evolution, evolution process, program evolution dynamics, software maintenance of software and legacy system management. Preliminary risk analysis identifies risks from the systems environment. Aim is to develop an initial set of system *security* and *dependability* requirements.

Unit-4

Sociotechnical systems organizational development an approach to complex organizational work design that recognizes the interaction between people and technology in workplaces. *Systems engineering* management that focuses on how to design and manage complex *systems* over their life cycles. Operation of system. Dependability and security their properties and uses. Risk driven requirements, specification reliability specification, security specification, safety specification and formal specification.

Unit-5

Dependability and *Computer Engineering*: Concepts for Software-Intensive Systems offers a state-of-the-art overview of the *dependability* research. Redundancy is generally applied to hardware and components and is intended to make the full system more resilient. Diversity is a specific instance of redundancy. Dependable systems architecture, dependable programming and security engineering. Security mass management needs of organization. Security survivability, dependability and security assurance. Software Reliability is the probability of failure-free software operation for a specified period of time in a specified environment. Process assurance and safety dependability cases.

DSE:1-(Information Security)

Unit1-

Cryptography is an indispensable tool for protecting information in computer systems. In this course you will learn the inner workings of cryptographic systems and how to correctly use them in real-world applications.

Unit2-

Being human, programmers and other developers make many mistakes, most of which are unintentional and not malicious. Many such errors cause program malfunctions but do not lead to more serious security vulnerabilities. However, a few classes of errors have plagued programmers and security professionals for decades, and there is no reason to believe they will disappear. In this section we consider three classic error types that have enabled many recent security breaches.

Unit3-

Database security concerns the use of a broad range of information security controls to protect databases (potentially including the data, the database applications or stored functions, the database systems, the database servers and the associated network links) against compromises of their confidentiality, integrity and availability. *Database security* is a specialist topic within the broader realms of computer security, information security and risk management.

Unit4-

Network security combines multiple layers of defenses at the edge and in the network. Each network security layer implements policies and controls. Authorized users gain access to network resources, but malicious actors are blocked from carrying out exploits and threats.

Unit5-

An information security strategic plan can position an organization to mitigate, transfer, accept or avoid information risk related to people, processes and technologies. Protecting sensitive data is the end goal of almost all IT security measures. Two strong arguments for protecting sensitive data are to **avoid** identity theft and to **protect privacy**.

DSE:2-(Microprocessor)

Unit -1

Processor architecture is the science of integrating those components to achieve a level of functionality and performance. **MU0 an abstract design, The study of MU0 processor designing, processor design trade-offs. Design of low power consumption.** ARM processor is one of a family of CPUs based on the RISC architecture developed by Advanced RISC Machines. The arm programmers' model and arm development tools.

Unit-2

General data processing instructions. Add, Subtract, and Reverse Subtract. Data transfer instructions move data between memory and the general-purpose and segment registers. Control flow instructions, arm organization and implementation. 3-stage pipeline ARM organization, ARM coprocessor interface.

Unit-3

Conditionally execute statement lists according to the value of a Boolean expression. Branch and *exchange* instruction set, Branch with *Link and exchange* instruction set. Software interrupt, data processing instruction. *Multiple register transfer instructions* provide efficient way of moving the contents of several registers. *Coprocessor Register Transfer Instructions*. LDC and STC Transfer data between memory and coprocessor. Breakpoint instruction, unused instruction, space and memory faults

Unit-4

An abstraction during design gives the designer freedom to field its explicit formulation. Data types like integer, float, double. Conditional statement of program, loops programming, function and procedures. Memory uses when and where, runtime environment and program examples.

Unit-5

The *Thumb* instruction set consists of 16-bit instructions. Thumb branch instruction. Thumb software interrupts instruction. *Thumb general data processing instructions* .thumb data processing instruction contains the following subsections: AND, ORR, EOR, and BIC Bitwise logical operations. Thumb single register data transfer instructions for accessing half-words were a later addition to the instruction set so have some restrictions. Thumb multiple register data transfer instruction. Thumb breakpoint instruction, thumb implementation, thumb application, ARM memory interface. Bus communication system that transfers data between components inside a computer, or between computers. ARM reference peripheral specification. ARM Instruction Set Simulator, *ARMulator*, is one of the software development tools provided by the development systems business unit.

SEMESTER-VI

C:13-(Artificial Intelligence)

Unit1-

In artificial intelligence, an intelligent agent (IA) is an autonomous entity which observes through sensors and acts upon an environment using actuators (i.e. it is an agent) and directs its activity towards achieving goals (i.e. it is "rational", as defined in economics). Search is a central topic in Artificial Intelligence. This part of the course will show why search is such an important topic, present a general approach to representing problems to do with search, introduce several search algorithms

Unit2-

Knowledge representation and reasoning (KR) is the field of artificial intelligence (AI) dedicated to representing information about the world in a form that a computer system can utilize to solve complex tasks such as diagnosing a medical condition or having a dialog in a natural language.

Unit 3-

Most real-world Planning problems are multi-objective, trying to minimize both the make span of the solution plan, and some cost of the actions involved in the plan.

Unit 4-

Decision tree learning is one of the most successful techniques for supervised classification learning. In this all of the features have finite discrete domains, and there is a single target feature called the classification. The idea of ANNs is based on the belief that working of human brain by making the right connections can be imitated using silicon and wires as living neurons and dendrites. Reinforcement learning (RL) is an area of machine learning inspired by behaviorist psychology, concerned with how agent sought to take actions in an environment so as to maximize some notion of cumulative reward.

Unit5-

Natural language processing (NLP) is a field of computer science, artificial intelligence and computational linguistics concerned with the interactions between computers and human (natural) languages, and, in particular, concerned with programming computers to fruitfully process large natural language corporation.

C:14-(Design And Analysis Of Algorithms)

Unit-1

Uses of Analysis and design of algorithm. *Asymptotic Analysis* that handles above issues in analyzing algorithms. Divide and conquer algorithm design paradigm based on multi-branched recursion. *Recurrence relation* recursively defines sequence or multidimensional array of values. *Strassen's matrix* Divide and Conquer method to multiply two matrices.

Unit-2

Quick Sort is a Divide and Conquer algorithm. *Heap sort* is a comparison based sorting technique based on Binary Heap data structure. Counting sort technique based on keys between specific ranges. Randomized quick sort of integer or data. Order statistics.

Unit-3

Amortized analysis is a method for analyzing a given algorithm's time complexity. 2–3–4 tree self-balancing data structure, commonly used to implement dictionaries. Fibonacci heap data structure for priority queue operations. Black color node. Hashing to map data of arbitrary size to data of fixed size. *Succinct data structure* uses amount of space to the information-theoretic lower bound.

Unit-4

Matrix chain multiplication and optimization problem that can be solved using dynamic programming. *Algorithm* code returns only length of *LCS*. Travelling Salesman Problem (*TSP*). *Branch and bound* algorithm design paradigm for discrete and combinatorial optimization problems. Greedy algorithm paradigm follows the problem solving heuristic of making the locally optimal choice at each stage with the hope of finding a global optimum. Study of Convex hull, fractional knapsack and back tracking.

Unit-5

PSPACE is the set of all decision problems. NP-complete decision problem is one belonging to both the NP and the NP-hard complexity classes. Vertex cover of a graph is a set of vertices. *Graph coloring* problem to assign colors to certain elements of a graph subject to certain constraints. Uses of cycle matching, exact cover, independent set, Hamiltonian and cheque. Minimum size *vertex cover* of a graph is NP complete, Travelling Salesman Problem (*TSP*), and *sum of subset*.

DSE:3-(Cloud Computing)

Unit-1

Grid computing collection of *computer* resources from multiple locations to reach a common goal. *Computer cluster* set of loosely or tightly connected *computers* that work together. *Distributed computing* systems are shared among multiple *computers*. *Cloud computing* paradigm that enables ubiquitous access to shared pools of configurable system. History of cloud computing and introduction too. Cloud service provider and benefits.

Unit-2

A comparison of *traditional* methods versus current *cloud computing*. *Infrastructure as a service (IaaS)* refers to online *services* that provide high-level APIs used according to the Internet Engineering Task Force. Enterprise *PaaS* provides software developers self-service portal for managing computing *infrastructure*. *SaaS* users are responsible for managing applications, data, runtime, middleware. Works of cloud computing and deployment. Public cloud standard cloud computing model, private cloud, hybrid cloud, community cloud.

Unit-3

Case study of services, The *Model Class application* describes the kinds of data it uses *with models*. *Microsoft Azure* open, flexible, enterprise-grade cloud computing platform uses. Compute Cloud *Amazon EC2* web service provides secure, resizable compute capacity in the cloud.

Unit-4

Cloud service management and *cloud* monitoring encompass a broad range of processes and practices. *Service-level agreement (SLA)* official commitment between a *service* provider and client. Billing and *accounting* programs of document financial *accounting*. Comparing scaling hardware and economics of scaling. Core difference between traditional IT and the cloud.

Unit-5

Cloud security refers to a broad set of policies, technologies, and controls deployed to protect data, applications, host level security, application level security, and network level security. Protective digital privacy to prevent unauthorized access to computers, databases and websites. Security issues, jurisdictional issues and data privacy. Identity management and *authentication* form the basis for security in *cloud* or on the local network.

DSE:4-(PROJECT WORK)

The project provides an important opportunity for students to plan and carry out a detailed and original piece of scientific research and communicate the results. You should develop the following abilities:

1. The formulation of scientific questions, the planning of an investigation and the design of individual experiments.
2. In-depth scientific review of a subject.
3. Organisation of research including: logistics, recording, archiving, numerical analysis and presentation of data.
4. Technical expertise.
5. Interpretation and presentation of results in the form of a dissertation.

DEPARTMENT OF I.T & COMPUTER SCIENCE**SEMESTER-I**

CORE	PAPER NAME	OBJECTIVE
CORE-1	C:1-PROGRAMMING USING C	<p>At its core, Objective-C is an ANSI standard version of the C programming language. Wrapped around this ANSI C core is a Smalltalk-inspired set of extensions that give the language its object-oriented capabilities, as well as several other enhancements that you don't get from the regular version of C.</p> <p>Brad Cox and Tom Love created the Objective-C programming language in the early 1980s in an effort to get people to write cleaner, more modular, and clearly separated code. Contrary to popular belief, Objective-C wasn't invented by, nor is it exclusively owned by, Apple. It's actually an open standard; in the past, implementations of the Objective-C compiler existed that even ran on Windows.</p> <p>If you have had any experience with C, learning Objective-C should be a breeze. Most developers find that learning the Objective-C syntax takes very little time at all, and the rest of the learning curve is devoted to learning about all the tools and controls available in Cocoa for Mac OS X.</p>
CORE-2	C:2-COMPUTER ORGANIZATION	<ul style="list-style-type: none">• To apply the knowledge of performance metrics to find the performance of systems.• To create an assembly language program to program a microprocessor system.• To design a hardware component for an embedded system• To deal with different types of computers• To identify high performance architecture design• To identify the problems in components of computer.• To develop independent learning skills and be able to learn more about different computer architectures and hardware.• To learn & use the new technologies in computers.• To use the knowledge of micro programming in the field of speech processing.

--	--	--

SEMESTER-II		
CORE	PAPER NAME	OBJECTIVE
CORE-3	C:1-PROGRAMMING USING C++	<p>The prime purpose of C++ programming was to add object orientation to the C programming language, which is in itself one of the most powerful programming languages.</p> <p>The core of the pure object-oriented programming is to create an object, in code, that has certain properties and methods. While designing C++ modules, we try to see whole world in the form of objects. For example a car is an object which has certain properties such as color, number of doors, and the like. It also has certain methods such as accelerate, brake, and so on.</p>
CORE-4	C:4-DATA STRUCTURE	<ul style="list-style-type: none"> • To apply the knowledge of performance metrics to find the performance of systems. • To create an assembly language program to program a microprocessor system. • To design a hardware component for an embedded system • To deal with different types of computers • To identify high performance architecture design • To identify the problems in components of computer. • To develop independent learning skills and be able to learn more about different computer architectures and hardware. • To learn & use the new technologies in computers. • To use the knowledge of micro programming in the field of speech processing.

SEMESTER-III

CORE	PAPER NAME	OBJECTIVE
CORE-5	C:5-OPERATING SYSTEM	<p>An operating system is a program that acts as an interface between the software and the computer hardware.</p> <ul style="list-style-type: none">• It is an integrated set of specialized programs used to manage overall resources and operations of the computer. <p>It is a specialized software that controls and monitors the execution of all other programs that reside in the computer, including application programs and other system software</p> <p>The objectives of the operating system are –</p> <ul style="list-style-type: none">• To make the computer system convenient to use in an efficient manner.• To hide the details of the hardware resources from the users.• To provide users a convenient interface to use the computer system.• To act as an intermediary between the hardware and its users, making it easier for the users to access and use other resources.• To manage the resources of a computer system.• To keep track of who is using which resource, granting resource requests, and mediating conflicting requests from different programs and users.• To provide efficient and fair sharing of resources among users and programs.
CORE-6	C:6-DATABASE MANAGEMENT SYSTEM	<ol style="list-style-type: none">1. Understand the areas of database design, SQL and programming2. Understand relational and object oriented database technology for building applications for the current trend3. Evaluate a business situation and designing & building a database applications

		<ul style="list-style-type: none"> • List and explain the fundamental concepts of a relational database system. • Utilize a wide range of features available in a DBMS package. • Analyze database requirements and determine the entities involved in the system and their relationship to one another. • Develop the logical design of the database using data modeling concepts such as entity-relationship diagrams. • Create a relational database using a relational database package. • Manipulate a database using SQL. • Assess the quality and ease of use of data modeling and diagramming tools. <ul style="list-style-type: none"> • To deal with different types of computers • To identify high performance architecture design • To identify the problems in components of computer. • To develop independent learning skills and be able to learn more about different computer architectures and hardware. • To learn & use the new technologies in computers. • To use the knowledge of micro programming in the field of speech processing.
CORE-7	C7:- DISCRETE STRUCTURES	<p>Objective 1: Students will learn basic logic and set theory.</p> <p>Objective 2: Students will learn core ideas in combinatorial mathematics.</p> <p>Objective 3: Students will learn core ideas in graph theory. Evaluate Boolean functions and simplify expression using the properties of Boolean algebra ,apply Boolean algebra to circuits and gating networks.</p> <p>Use finite-state machines to model computer operations.</p>

SEMESTER-IV

CORE	PAPER NAME	OBJECTIVE
CORE-8	C:8-JAVA PROGRAMMING	<ul style="list-style-type: none"> • This course provides an introduction to object oriented programming (OOP) using the Java programming language. <ul style="list-style-type: none"> ○ Its main objective is to teach the basic concepts and techniques which form the object oriented programming paradigm • Students completing the course should know: <ul style="list-style-type: none"> ○ The model of object oriented programming: abstract data types, encapsulation, inheritance and polymorphism ○ Fundamental features of an object oriented language like Java: object classes and interfaces, exceptions and libraries of object collections ○ How to take the statement of a business problem and from this determine suitable logic for solving the problem; then be able to proceed to code that logic as a program written in Java. ○ How to test, document and prepare a professional looking package for each business project using javadoc.
CORE-9	C:9-COMPUTER NETWORK	<p>A computer network is defined as interconnected collection of autonomous computers. Computer are said to be interconnected, if they able to exchange information. Connection is physically established through cables, lasers, microwaves, fiber optics and communication satellite.</p> <p>Right from the day of computer automation, a computer has developed computer in each department. For example, the task can be to keep of track off inventories, monitor productivity and maintain accounts. Initially, this computer in different department works as an isolation from other computer. These provided the necessary computer assistance in the activities pertaining to the</p>

		<p>respective department. It was not possible to share information among the departments. As a result, the information was replicated wherever needed. This increased redundancy caused increased in storage space, less data consistency etc.</p> <p>Hence, at a point it was decided to connect the computers in various departments to extract the information and correlate information about the entire company.</p>
CORE-10	C10:- COMPUTER GRAPHICS	<p>Computer Graphics is the illustration field of Computer Science. Its use today spans virtually all scientific fields and is utilized for design, presentation, education and training. Computer Graphics and its derivative, <i>visualization</i>, have become the primary tools by which the flood of information from Computational Science is analyzed.</p> <ul style="list-style-type: none"> • To Identify and explain the core concepts of computer graphics. • Apply graphics programming techniques to design, and create computer graphics scenes. • Create effective OpenGL programs to solve graphics programming issues, including 3D transformation, objects modeling, colour modeling, lighting, textures, and ray tracing.
SEC:II	SEC:II-ANDROID PROGRAMMING	<p>Build and deploy his/ her Android application.</p> <ul style="list-style-type: none"> • Students understand the operation of the application, application lifecycle, configuration files, intents, and activities. • The candidates get a better understanding of the UI - components, layouts, event handling, and screen orientation. • Students also develop a working knowledge of the custom UI elements and positioning. • The candidates may also have an in-depth understanding of broadcast receivers and services. • The networking capabilities such as JAVA Sockets, JAVA XML and JSON are taught. • The trainee may develop a basic application that acts as a working example of all the topics covered in the class.

SEMESTER-V

CORE	PAPER NAME	OBJECTIVE
CORE-11	C:8-INTERNET TECHNOLOGY	<p>The aim of this course is to provide you the conceptual and technological developments in the field of Internet and web designing with the emphasis on comprehensive knowledge of Internet, its applications and the TCP/IP protocols widely deployed to provide Internet connective worldwide. The World Wide Web with its widespread usefulness has become an integral part of the Internet. Therefore, this course also puts emphasis on basic concepts of web design.</p> <ul style="list-style-type: none">• Review the current topics in Web & Internet technologies.• Describe the basic concepts for network implementation.• Learn the basic working scheme of the Internet and World Wide Web• Understand fundamental tools and technologies for web design.• Comprehend the technologies for Hypertext Mark-up Language (HTML).• Specify design rules in constructing web pages and sites.• Effectively deal with programming issues relating to VB Script, JavaScript, Java, ASP, Front Page and Flash.• Figure out the various security hazards on the Internet and need of security measures.

CORE-12	C:12-SOFTWARE ENGINEERING	<ol style="list-style-type: none"> 1. Maintainability – the ease with which changes in a functional unit can be performed in order to meet prescribed requirements. 2. Correctness – the extent to which software meets its specified requirements 3. Reusability – the extent to which a module can be used in multiple applications. 4. Testability – the extent to which software facilitates both the establishment of test criteria and the evaluation of the software with respect to those criteria. 5. Reliability – an attribute of software quality. The extent to which a program can be expected to perform its intended function, over an arbitrary time period. 6. Portability – the ease with which software can be transferred from one computer system or environment to another. 7. Adaptability – the ease with which software allows differing system constraints and user needs to be satisfied by making changes to the software.
DSE:1	DSE:1-INFORMATION SECURITY	<ol style="list-style-type: none"> 1. Educational Objectives: This introductory course is aimed at giving basic understanding about system security. This entry-level course covers a broad spectrum of security topics and is based on real-life examples to create system security interest in the students. A balanced mix of technical and managerial issues makes this course appealing to attendees who need to understand the salient facets of information security basics and the basics of risk management. 2. Course Outcomes: At the end of the course, the students have firm understanding on basic terminology and concepts related to network and system level security, basics of computers and networking including Internet Protocol, routing, Domain Name Service, and network devices. They are also exposed to basic cryptography, security management, and network security techniques. They also look at policies as a tool to effectively change an organization's culture towards a better secure environment.

		<p>In the end, the students put it all together in the form of a case study for designing and auditing a security system at conceptual level.</p>
<p>DSE:2</p>	<p>DSE:2-MICROPROCESSOR</p>	<p>A microprocessor is a computer processor which incorporates the functions of a computer's central processing unit (CPU) on a single integrated circuit (IC), or at most a few integrated circuits. The microprocessor is a multipurpose, clock driven, register based, digital-integrated circuit which accepts binary data as input, processes it according to instructions stored in its memory, and provides results as output. Microprocessors contain both combinational logic and sequential digital logic. Microprocessors operate on numbers and symbols represented in the binary numeral system.</p> <p>The integration of a whole CPU onto a single chip or on a few chips greatly reduced the cost of processing power, increasing efficiency. Integrated circuit processors are produced in large numbers by highly automated processes resulting in a low per unit cost. Single-chip processors increase reliability as there are many fewer electrical connections to fail. As microprocessor designs get better, the cost of manufacturing a chip (with smaller components built on a semiconductor chip the same size) generally stays the same.</p>

SEMESTER-VI

CORE	PAPER NAME	OBJECTIVE
CORE-13	C:13-ARTIFICIAL INTELLIGENCE	<p>The objective of the course is to present an overview of artificial intelligence (AI) principles and approaches. Develop a basic understanding of the building blocks of AI as presented in terms of intelligent agents: Search, Knowledge representation, inference, logic, and learning. Students will implement a small AI system in a team environment. The knowledge of artificial intelligence plays a considerable role in some applications students develop for courses in the program.</p> <p>Learning Outcomes:</p> <p>Upon successful completion of this course student will:</p> <ul style="list-style-type: none">- be able to design a knowledge based system,- be familiar with terminology used in this topical area,- have read and analyzed important historical and current trends addressing artificial intelligence.
CORE-14	C:14-DESIGN AND ANALYSIS OF ALGORITHMS	<p>The objective of the course is to teach techniques for effective problem solving in computing. The use of different paradigms of problem solving will be used to illustrate clever and efficient ways to solve a given problem. In each case emphasis will be placed on rigorously proving correctness of the algorithm. In addition, the analysis of the algorithm will be used to show the efficiency of the algorithm over the naive techniques.</p>
DSE:-3	DSE:3-CLOUD COMPUTING	<p>The student will learn about the cloud environment, building software systems and components that scale to millions of users in modern internet, cloud concepts capabilities across the various cloud service models including Iaas,Paas,Saas, and developing cloud based software applications on top of cloud platforms.</p> <ol style="list-style-type: none">1. Understanding the key dimensions of the challenge of Cloud Computing2. Assessment of the economics , financial, and technological implications for selecting cloud computing for

		<p>own organization</p> <p>3. Assessing the financial, technological, and organizational capacity of employer's for actively initiating and installing cloud-based applications.</p> <p>4. Assessment of own organizations' needs for capacity building and training in cloud computing-related IT areas.</p>
DSE:-4	DSE:4- PROJECT WORK	<p>The project provides an important opportunity for students to plan and carry out a detailed and original piece of scientific research and communicate the results. You should develop the following abilities:</p> <ol style="list-style-type: none"> 1. The formulation of scientific questions, the planning of an investigation and the design of individual experiments. 2. In-depth scientific review of a subject. 3. Organisation of research including: logistics, recording, archiving, numerical analysis and presentation of data. 4. Technical expertise. 5. Interpretation and presentation of results in the form of a dissertation.

