

DEPARTMENT OF SOFTWARE ENGINEERING

B.TECH. 1st Year Syllabus

1st Sem

Details of Course:

Course Title: Skill Enhancement Course 1

Course Title	Course Structure			Pre-Requisite
Computer Workshop 1	L	T	P	Nil
	0	0	4	

Course Objective:

Students of Software Engineering are to work with various hardware and software not only in academia but also in the company. Thus, students should get familiar with various hardware, software, operating systems, and networking. This course will provide students a much-needed knowledge of computer hardware and networking, enabling them to identify and rectify onboard computer hardware, software, and network-related problems. With the help of this course, the student will be able to understand the hardware specifications that are required to run an operating system and various application programs.

Course Outcome (CO):

After completing their training in Computer Workshop 1, students will be able to

CO1. Describe the procedure for installation of software on different systems and identify the various components of hardware systems.

CO2. Identify and demonstrate components of computer and operating system and their troubleshooting.

CO3. Describe the basics of Internet and web design

CO4. Perform the process of software installation

S.No.	Content	Contact Hours
Unit 1	Assembly/Disassembly of Computers: Hardware peripherals like RAM, ROM, input devices, output devices, processors, etc. Processors and processor core counts and frequency etc. motherboards, internal and external connectors. Types of data cables. LAN, Audio, and Video. The physical set-up of Printers- Scanner set-up, Webcam, Bluetooth device, Memory card reader, etc.	10

	<p>Working of SMPS. Connection of different types of devices to the ports (CPU), Single board computer: Raspberry Pi.</p> <p>Assembly/Dis-assembly of Laptop: Mounting of processor. Fixing of the motherboard in the tower case. Connection to the power supply. Installation of drivers. Connection of cables. Mount the memory modules. Install the internal cards. Connection of the external devices and power.</p>	
Unit 2	<p>Computer Network Setup: Networking components, devices, and tools; Preparing the network cables, network setup, configuration and management commands, Installation and configuration of network interface card and identification of MAC address. Sharing of resources</p> <p>Software Installations: Installation of Windows Operating System, Types of software and their installations, some useful software (MS office, Adobe Acrobat, Google Chrome, VLC Media Player, LibreOffice, Win Rar)</p>	12
Unit 3	<p>PC Maintenance: POST (Power on Self-Test), identifying problems by Beep codes errors, checking power supply using Multi-meter, Replacement of components etc.</p> <p>Introduction to MS office: Introduction to MS office - MS Word, MS PPT, MS Excel, Working with MS Word.</p>	12
Unit 4	<p>Tools for Online Teaching and Meetings: Setting & troubleshooting of online meetings and video conferencing like google meet, zoom, Microsoft teams, Webex etc; use of google classroom and google forms for teaching, feedback, and evaluation.</p> <p>Internet and Basic Webpage Design: Searching the Internet, checking the speed of Internet connection, usage of E-Commerce, Creating webpage using HTML, CSS with static text, images, tables, audio, video etc and dynamic contents, animation usage and tools for webpages</p>	10
Unit 5	<p>AI & ML Applications: Case studies using module (Blackbox based) integration for AI & ML and its applications</p>	12
	Total	56

2nd Sem

Details of Course:

Departmental Core Course

Course Title: Fundamentals of Software Engineering

Course Title	Course Structure			Pre-Requisite
Fundamentals of Software Engineering	L	T	P	Nil
	4	0	0	

Course Objective:

Students of Software Engineering are to work with software in the company. Students should get familiar with the basic knowledge of software engineering for developing a product. This course will provide students a sufficient knowledge of software development models, categories of software requirements, criterion for designing an efficient software. Thus, this course will enable students to develop a software according to user requirements using appropriate software development process.

Course Outcome (CO):

After completing their course in Fundamentals of Software Engineering, students will be able to

CO1. Describe the phases of software development life cycle for designing an efficient software.

CO2. Identification of user requirements using various requirements elicitation techniques.

CO3. Describe the procedure of designing software requirement specification for designing software as per user requirements.

CO4. Describe the basics of software design using various techniques.

S.No.	Content	Contact Hours
Unit 1	Introduction: Software, Program, Software Crisis and Solutions, Software Evolution, Software Development Paradigm, Need of Software Engineering, Software Processes, Software Characteristics, Attributes of Software Product, Software Myths, Software basic terminologies, Characteristics of good software.	10
Unit 2	Software life cycle models: Build and Fix, Waterfall, Prototype, Iterative Enhancement, Rapid Application Development, Evolutionary, Prototyping, and Spiral Model.	12
Unit 3	Software Requirements Analysis and Specifications: Requirement Engineering, Types of Requirements. Requirement Elicitation: Interview, Brainstorming, Quality Functional Deployment, Use Case	10

	Approach.	
Unit 4	Requirements Analysis: Problem Analysis, Data Flow Diagrams, Data Dictionaries, Entity-Relationship Diagram, Requirements Documentation, Requirements Validation, Software Requirement and Specifications, Requirements Management, Change Management Form, Structure of SRS, IEEE Std 830-1993, Software Prototyping.	12
Unit 5	Software Design: Design framework, Conceptual and Technical Design, Trade-off between modularity and software cost, Cohesion and Coupling, Types of cohesion and coupling, strategy of design, Structure chart, IEEE standard 1016-1998 for Software Design Description.	12
	Total	56

Books:-

S.No.	Name of Books/Authors/Publisher
1.	Software Engineering, K. K. Aggarwal and Yogesh Singh, New Age International Private Limited; Fourth edition.
2.	Software Engineering – A practitioner’s approach, R. S. Pressman, 6th ed., McGraw Hill Int. Ed.
3.	Software Engineering Concepts, Richard Fairley, McGraw Hill Education.
4.	An Integrated Approach to Software Engineering, Pankaj Jalote, Narosa.
5.	Software Engineering, Ian Sommerville, Pearson Education; Tenth edition.

Details of Course:**Interdisciplinary Core Course - 3****Course Title: Fundamentals of Computers**

Course Title	Course Structure			Pre-Requisite
Fundamentals of Computers	L	T	P	Nil
	3	0	2	

Course Objective:

Everyday, engineering students required to work with computer for problem solving in academia, research, and industry. Students will get knowledge about the evolution of computer, computer architecture, input devices, output devices, computer codes, computer software, operating system, computer arithmetic, and internet. Thus, students will be able to learn about the importance of data storage in the computer and computer security. With the help of this course, the student will be able to understand the functioning of computer units, and usage of computer to do everyday task.

Course Outcome (CO):

After completing their course in Fundamentals of Computer, students will be able to

CO1. Describe the procedure of designing algorithm and drafting pseudocode for problem solving.

CO2. Describe the computer organization and architecture of central processing unit.

CO3. Describe the computer codes, computer arithmetic and number conversion system.

CO4. Describes the procedure of installing functionalities and installation of different operating system, software.

CO5. Describe the process of data storage and modelling using database management system.

S.No.	Content	Contact Hours
Unit 1	Introduction: Evolution of Computers, Generation of Computers, Classification of Computers, Computing Concepts, The Computer System, Applications of Computers. Computer Organization and Architecture: Central Processing Unit, Internal Communications, Machine Cycle, The Bus and Instruction Set.	8
Unit 2	Memory and Storage Systems: Memory Representation, Random Access Memory, Read Only Memory, Storage System - Magnetic, Optical, Magneto, Solid State, Storage	8

	evaluation criteria. Input Devices. Output Devices	
Unit 3	Computer Codes: Decimal System, Binary System, Hexadecimal system, Octal System, 4-bit Binary Coded Decimal, 8-bit BCD System, 16-bit Unicode, Conversion of Numbers. Computer Arithmetic. Binary Systems, Boolean Algebra and logic gates (AND, OR, NOR, and XOR), Simplification of Boolean functions, combinational logic, adders, subtractors, code conversions, synchronous and asynchronous sequential logic.	8
Unit 4	Computer Software: Types of Computer Software, System Management Program, System Development Programs, Standard Application Programs. Operating System: History of Operating System, Functions of Operating System, Process Management, Memory Management, File Management, Device Management, Security Management, Types of Operating System.	8
Unit 5	Database Management System: Database system, File System, Data Models, Data Independence, DBMS System Architecture, Components of DBMS, Relational Data Model, Relational Database Design, Data Storage and Indexing. Computer Security: Overview of Security Threats and Vulnerability, Malware-viruses, worms, trojan horse. Security counter measures, Intrusion Detection, Antivirus Software, Information Security, Privacy, Ethical Issues, Ethical Hacking. Ethical issues in information security and privacy	10
	Total	42

Books:-

S.No.	Name of Books/Authors/Publisher
1.	Fundamentals of Computer Science, Balaguruswamy, McGraw Hill Education (India) Private Ltd.
2.	Fundamentals of Computers, V. Rajaraman, PHI
3.	Handbook of Computer Fundamentals, Nasib Singh, Khanna Books Publishing Co. (P) Ltd.
4.	Computer Fundamentals, P.K. Sinha, BPB Publication.
5.	Introduction to Computer, Norton Peter, McGraw-Hill.

Details of Course:

Course Title: Skill Enhancement Course 2

Course Title	Course Structure			Pre-Requisite
Computer Workshop 2	L	T	P	Nil
	0	0	4	

Course Objective:

Students of Software Engineering require to develop software or product for solving real world problems in academia, and industry. Thus, this course will teach the process of developing a software with feasible solution. Students will gain knowledge about storing the data in a system, using diagrammatic representation and establishing relationship among different attributes of a data. This course will help them in understanding diagrammatically the flow of data among different modules. With the help of this course, students will be able to understand the importance of analyzing problem and it's solution from developer and customer perspective.

Course Outcome (CO):

After completing their training in Computer Workshop 2, students will be able to

- CO1. Demonstrate the modelling of data stored in a database.
- CO2. Demonstrate the way information is flowing through the system.
- CO3. Describe the procedure for designing data flow diagram and context diagram.
- CO4. Describe the process of interaction among external entities with an internal software system.
- CO5. Demonstrate the process of collecting requirements form the user for software development.

S.No.	Content	Contact Hours
Unit 1	Software Requirement: Development of Software Requirement Specification, Software Requirements Analysing tools, Validation of Requirements, Case study based on Requirement Specification Engineering.	12
Unit 2	Data Flow Diagrams: Symbols used for constructing DFD, Synchronous and Asynchronous Operations, Data Dictionary, DFD model of a system consisting of hierarchy of DFDs, construction of context diagram, construction of levell diagram, construction of lower-level diagrams, construction of level 2 diagrams, data dictionary for the DFD model.	12
Unit 3	Structure Chart: Extension of DFD technique for designing real-time systems, Structured design,	10

	transformation of a DFD Model into structure chart, transform analysis, transaction analysis, and detailed design.	
Unit 4	<p>Microsoft Excel: Manage workbook options and settings, apply custom data formats and layouts, create tables, perform operations with formulas and functions, create charts and objects, manage workbook options and settings, apply advanced conditional formatting and filtering, prepare a workbook for internationalization, create advanced formula, perform data analysis, troubleshoot problems, create and manage pivot tables, create and manage pivot charts.</p> <p>Introduction to Structured Query Language – SQL Data Types, SQL Operators, SQL Expressions, SQM Comments, Data Definition Language, Data Manipulation Language, Data Control Language, SQL Functions, SQL Queries and Sub Queries, Case study based on SQL Queries.</p>	12
Unit 5	Microsoft PowerPoint: Introduction, windows features, presentation slides, create slide presentation, editing techniques, slide master, format slide, transitions and animation, slide illustration and shapes, slide show, print presentation.	10
	Total	56

DEPARTMENT OF SOFTWARE ENGINEERING

B.Tech. 2nd Year Syllabus

Course Title	Course Structure			Pre-Requisite
Digital Systems & Design	L	T	P	Concepts of Boolean Algebra, basic Logic gates
	3	0	2	

Course Objective:

To introduce the concepts of digital logic, functioning and design of digital devices, Programmable Devices, memory, and digital system design using VHDL.

Course Outcome (CO):

1. Apply knowledge of minimization techniques to switching functions, and realization of FSM.
2. Apply synchronous sequential logic concept for designing Finite state machines.
3. Apply Asynchronous sequential logic concepts for designing circuits from given statements and apply ASM concepts for designing digital circuits.
4. Understand the concept of various ADC and DAC conversion and various techniques and designing circuits using programmable logic devices.
5. Understand the concept of HDL and demonstrate its knowledge by designing various digital systems.
6. Understand the concept of various logic families and their parameters.

S.No.	Content	Contact Hours
Unit 1	Review of basic concepts of switching functions, and minimization techniques (Karnaugh's Map Method & Tabulation Techniques). Introduction to finite state machine: pulse and fundamental mode of operation, realization of state table from verbal description, state diagram & Transition matrix, Mealy and Moore model machine.	8
Unit 2	Design of sequential circuits: Flip Flops, Shift Registers, Ring counters, Up-Down counters, Asynchronous counters, decade counters.	8
Unit 3	Introduction to design asynchronous sequential circuit flow table realization from verbal description, ASM charts.	8
Unit 4	Concept of Digital to Analog Conversion Ladder Networks, and Concept of Analog to Digital conversion: Dual Slope method, stair case Ramp-method/counter method successive approximation type of A/D converters etc. Introduction to design with the programmable modules: ROM, PAL, PLA, FPGA .	8
Unit 5	Introduction to HDL (VHDL), Behavioral	6

	Modeling, Dataflow Modeling, Structural Modeling, and Application in Digital System Designs.	
Unit 6	Introduction Logic Gates Families TTL, Tristate Logic, ECL, CMOS and I ² L Logic, Logic parameters, Bistable, Monostable, Astable and Schmitt trigger circuit	4
Total		42

Books:-

S.No.	Name of Books/Authors/Publisher
1.	Thomas L. Floyd, Digital Fundamentals, 10th Edition, Pearson Education, ISBN-13: 9780132359238, 2009.
2.	M. Morris Mano, Digital Design, 4th Edition, Pearson Education ISBN-13: 9780131989245, 2007.
3.	Donald P. Leach and Albert Paul Malvino, Goutam Saha, Digital Principles and Applications, 6th Edition, TMH, ISBN: 0070601755, 2006.
4.	John F. Wakerly, Digital Design: principles and practices, 4th Edition, Pearson Education, ISBN-10: 0131863894, 2006.
5.	John M. Yarbrough, Digital Logic Applications and Design, Thomson Learning, ISBN-10: 0314066756, 2002.
6.	Switching And Finite Automata Theory by Z. Kohavi (TMH). 2010
7.	VHDL Primer by J. Bhaskar; BS Publication. 2001

Course Title	Course Structure			Pre-Requisite
Data Structures	L	T	P	Fundamentals of Programming
	3	0	2	

Course Objective:

The objective of data structures and algorithms is to enable students to implement, understand, and analyze basic techniques of algorithm analysis. They define how efficiently we can store and organize data, as well as different algorithms for computational tasks. This subject aims to develop problem-solving skills, algorithmic thinking, and the ability to write efficient programs.

Course Outcome (CO):

1. Understand basic data structures such as arrays, linked list, stacks, and queues.
2. Analyze the concepts of algorithm evaluation, and find time and space complexities for searching and sorting algorithms.
3. Implement different types of trees, and apply them to problem solutions.
4. Discuss graph structure, and understand various operations on graphs and their applicability.
5. Apply algorithm for solving problems like sorting, searching, insertion, and deletion of data.

S.No.	Content	Contact Hours
Unit 1	<p>Introduction: Introduction to Algorithmic, Complexity- Time-Space Trade off. Introduction to abstract data types, design, implementation, and applications. Introduction to List data structure.</p> <p>Arrays and Strings: Representation of Arrays in Memory: one dimensional, Two dimensional, and Multidimensional, Accessing of elements of array, performing operations like Insertion, Deletion, and Searching. Sorting elements of arrays. Strings and String Operations.</p> <p>Stacks and Queues: Introduction to data structures like Stacks and Queues. Operations on Stacks and Queues, Array representation of Stacks, Applications of Stacks : recursion, Polish expression and their compilation conversion of infix expression to prefix and postfix expression, Operations of Queues, Representations of Queues Applications of Queues, Priority queues.</p>	9
Unit 2	<p>Linked Lists: Singly linked lists, Representation of linked list, Operations of Linked list such as Traversing, Insertion and Deletion, Searching, and Applications of Linked List. Concepts of</p>	8

	Circular linked list and Doubly linked list and their Applications. Stacks and Queues as linked list.	
Unit 3	Trees: Basic Terminology, Binary Trees and their representation, binary search trees, various operations on Binary search trees like traversing, searching, Insertion and Deletion, Applications of Binary Search Trees, Complete Binary trees, Extended binary trees. General trees, AVL trees, Threaded trees, B- trees.	8
Unit 4	Searching and Sorting: Linear Search, Binary Search, Interpolation Search, Insertion Sort, Quick sort, Merge sort, Heap sort, sorting on different keys, and External sorting.	8
Unit 5	Graphs: Terminology and Representations, Graphs & Multi-graphs, Directed Graphs, Representation of graphs and their Transversal, Spanning trees, shortest path and Transitive Closure, Activity Networks, Topological Sort and Critical Paths. File Structure: File Organization, Indexing & Hashing, Hash Functions, Collision Resolution Techniques.	9
Total		42

Books:-

S.No.	Name of Books/Authors/Publisher
1	Fundamentals of Data Structures; E. Horowitz, & S. Sahni, Universities Press, 2 nd Edition (2008).
2	Tannenbaum, "Data Structures", PHI
3	An introduction to data structures and application by Jean Paul Tremblay & Pal G. Sorenson (McGraw Hill).
4	R.L. Kruse, B.P. Leary, C.L. Tondo, "Data structure and program design in C", PHI

Course Title	Course Structure			Pre-Requisite
Object-Oriented Programming	L	T	P	Nil
	3	0	2	

Course Objective: To provide knowledge of Object-Oriented programming features.

Course Outcome (CO):

1. To apply mathematics to arrange and manipulate the data in a computational way
2. To differentiate between structured and object-oriented programming.
3. To apply concepts of Constructor, destructor, friend functions and classes & dynamic objects.
4. To apply concepts of polymorphism, inheritance, and abstraction in designing programs
5. Design, implement, test, debug, and document programs in C++
6. Analyze how the stack is used to implement function calls, and parameter passing options.
7. Write programs that perform explicit memory management.
8. Design template functions and classes for generic programming
9. Apply the knowledge of C++ programming in developing the application-oriented projects.

S.No.	Content	Contact Hours
Unit 1	Object oriented paradigm & C++ at a glance: Evolution of programming paradigm, structured versus object-oriented development, elements of object-oriented programming, Objects, classes, methods, popular OOP languages, software reuse. Classes and objects: Introduction, Class revisited, constant objects and constructor, static data members with constructors and destructors, constructor overloading, nested classes, objects as arguments, returning objects, friend functions and friend classes, constant parameters and member functions, static data and member functions.	8
Unit 2	Dynamic objects: Introduction, pointers to objects, array of objects, pointers to object members, this pointer, self-referential classes Operator overloading and Inheritance: overloading of new and delete operators, conversion between objects and basic types, conversion between objects of different classes, overloading with friend functions, abstract classes, inheritance types , virtual base classes, virtual functions, pointer to derived class objects, and base class objects, pure virtual functions, virtual destructors. Generic programming with templates: Introduction, function templates, overloaded function templates, class templates, inheritance of class template, class template containership, class template with overloaded operators.	7
Unit 3	Introduction to byte code, security and portability, Data Types, variables, operators, arrays, type conversion and	6

	casting, type promotion, Control statements, standard input-output, Designing Classes, constructors, methods, access specifiers: public, private, protected, inheritance, packages and interfaces, Math, String, Vectors, and Array List classes, polymorphism: function and operator overloading, function overriding, abstract classes.	
Unit 4	Exception Handling: exception types, nested try-catch, throw, throws and finally statements, Multithread Programming: thread creation, synchronization and priorities. Input-output and file operations: Java.io, stream classes, Byte streams, character streams, serialization. Networking concepts: Client server and socket programming, TCP/IP client and server sockets	13
Unit 6	Applets and Java Swing: Applet design, AWT packages, Applet event handling, parameters to applets, AWT controls, layout manager, Frames, container classes, Introduction to Java Beans, Swing and Servlets.	8
Total		42

Books:-

S.No.	Name of Books/Authors/Publisher
1	Patrick Naughton, Herbert Schildt: "The Complete Reference: Java 2", TMH.ISBN-13 9780070495432, 1999
2	C Thomas Wu: "An Introduction to OO programming with Java", TMH, ISBN-10: 0073523305, 2009
3	Balaguruswami, "Object oriented with C++", TMH. ISBN 0070669074, 9780070669079, 2008
4	Budd, "Object Oriented Programming", Addison Wesley, 1997
5	Mastering C++ K.R Venugopal Rajkumar, TMH, 2013
6	C++ Primer, "Lip man and Lajole", Addison Wesley, 1986
7	Maria litvin, Gary litvin, "Programming in C++", VPH, 2001
8	D Samantha, "Object oriented Programming in C++ and Java ", PHI, 2007

Course Title	Course Structure			Pre-Requisite
Operating System	L	T	P	Data Structures
	3	0	2	

Course Objective:

To familiarize with the fundamental principles of the operating system, its services and functionalities, the concepts of processes, synchronization and scheduling, memory management and need for protection in computer systems.

Course Outcome (CO):

1. Identify the basic concepts and functions of operating systems. Understand differentiation of various operating systems by their functionality.
2. Assess various process synchronization mechanisms and use different CPU scheduling methods in order to allocate resources effectively.
3. Understand various deadlock handling techniques to prevent and/or avoid deadlock.
4. Apply concepts of memory management including Virtual Memory and Page Replacement to the issues that occur in Real time applications.
5. Analyze the concepts related to file system interface, implementation, disk management, protection and security mechanisms

S.No.	Content	Contact Hours
Unit 1	Introduction: Operating system and function, Evolution of operating system, Batch, Interactive, Time Sharing and Real Time System, System protection. Operating System Structure: System Components, System structure, Operating System Services.	4
Unit 2	Concurrent Processes: Process concept, Principle of Concurrency, Producer Consumer Problem, Critical Section problem, Semaphores, Classical problems in Concurrency, Inter Process Communication, Process Generation, Process Scheduling. CPU Scheduling: Scheduling Concept, Performance Criteria Scheduling Algorithm, Evolution, Multiprocessor Scheduling.	9
Unit 3	Deadlock: System Model, Deadlock Characterization, Prevention, Avoidance and Detection, Recovery from deadlock combined approach.	8
Unit 4	Memory Management: Base machine, Resident monitor, Multiprogramming with fixed partition, Multiprogramming with variable partition, Multiple base register, Paging, Segmentation, Virtual memory concept, Demand paging, Performance, Paged replaced algorithm, Allocation of frames, Thrashing, Cache memory, Organization, Impact on performance.	9

Unit 5	I/O Management & Disk Scheduling: I/O devices and organization of I/O function, I/O Buffering, DISK I/O, Operating System Design Issues. File System: File Concept, File Organization and Access Mechanism, File Directories, File Sharing, Implementation Issues Case Studies: Windows, Linux and Unix	12
Total		42

Books:-

S.No.	Name of Books/Authors/Publisher
1	Silberschatz , Galvin, Gagne “Operating System Concepts”, Wiley, 9th Ed 2013
2	Tannaum, “Operating Systems”, PHI, 4th Edition 2000
3	Milnekovic, “Operating System Concepts”, McGraw Hill 1992
4	Harvey M Dietel, “ An Introduction to Operating System”, Pearson Education 2004

Course Title	Course Structure			Pre-Requisite
Software Engineering Methodologies	L	T	P	Fundamentals of SE
	3	0	2	

Course Objective:

To introduce the concepts of software engineering including requirement specifications, software design, testing and maintenance.

Course Outcome (CO):

1. Explain various software characteristics and analyse different software Development Models
2. Demonstrate the contents of a SRS and apply basic software quality assurance practices to ensure that design, development meet or exceed applicable standards
3. Compare and contrast various methods for software design
4. Formulate testing strategy for software systems, employ techniques such as unit testing, Test driven development and functional testing
5. Manage software development process independently as well as in teams and make use of Various software management tools for development, maintenance, and analysis

S.No.	Content	Contact Hours
Unit 1	Software Requirement Specification: Requirements Elicitation Techniques, Requirements analysis, Models for Requirements analysis, requirements specification, requirements validation.	9
Unit 2	System Design: Design Principles: Problem partitioning, abstraction. Top down and bottom up – design, structured approach. Functional versus 6object oriented approach of design, design specification, Cohesiveness and Coupling. Overview of SA/SD Methodology, structured analysis, data flow diagrams, extending DFD to structure chart.	8
Unit 3	Software project Management: Project planning and Project scheduling. Software Metrics: Size Metrics like LOC, Token Count, Function Count. Cost estimation using models like COCOMO. Risk management activities. Software Reliability and Quality Assurance: Reliability issues, Reliability metrics, reliability models, Software quality, ISO 9000 certification for software industry, SEI capability maturity model.	10
Unit 4	Testing: Verification and validation, code inspection, test plan, test case specification. Level of testing: Unit, Integration Testing, Top down and bottom-up integration testing, Alpha and Beta testing, System testing and debugging. functional testing, structural testing, Software testing strategies.	10
Unit 5	Software Maintenance: Structured Vs	5

	unstructured maintenance, Maintenance Models, Configuration Management, Reverse Engineering, Software Re-engineering. Case study: Practical applications of SDLC phases	
	Total	42

Books:-

S.No.	Name of Books/Authors/Publisher
1	R. S. Pressman, "Software Engineering – A practitioner's approach", 3rd ed., McGraw Hill Int. Ed..
2	K. K. Aggarwal & Yogesh Singh, "Software Engineering", 2ndEd., New Age International.
3	Sommerville, "Software Engineering", Addison Wesley.

4th Semester

Course Title	Course Structure			Pre- Requisite
Computer System Architecture	L	T	P	Digital Electronics
	3	0	2	

Course Objective:

To provide knowledge about the principles, concepts and applications of Computer Architecture.

Course Outcome (CO):

1. Describe the functionalities of various units of a computer
2. Illustrate the logic design of Control Unit
3. Understand the architecture and functionality of central processing unit.
4. Learn the different types of serial communication techniques.
5. Illustrate various memory components and memory mapping techniques

S.No.	Content	Contact Hours
Unit 1	Introduction: Digital computer generation, computer types and classifications, functional units and their interconnections, bus architecture, types of buses and bus arbitration. Register, bus and memory transfer. REGISTER TRANSFER LANGUAGE: Data movement around registers. Data transfer from/to memory, arithmetic, logic and shift micro operations. Concept of bus and timing in register transfer.	8
Unit 2	Control Unit: Instruction types, Instruction formats, Instruction cycles and sub-cycles (fetch and execute etc.), micro-operations, execution of a complete instruction. Hardwired Control Unit and Microprogrammed Control Unit: microprogrammed sequencing, wide branch addressing, and micro-instruction with next address field, pre-fetching microinstructions, concept of horizontal and vertical microprogramming.	8
Unit 3	Central Processing Unit: Processor organization, general register organization, stack organization and addressing modes. Computer Arithmetic: Addition and subtraction of signed numbers, Signed operand multiplication, Booth's algorithm and array multiplier. Division and logic operations. Floating point arithmetic operation	9

Unit 4	Input/Output organization: Peripheral devices, I/O interface, I/O ports, Interrupts: interrupt hardware, types of interrupts and exceptions. Modes of Data Transfer: Programmed I/O, interrupt initiated I/O and Direct Memory Access. I/O channels and processors. Serial Communication: Synchronous & asynchronous communication, standard communication interfaces.	9
Unit 5	Memory: Basic concept and hierarchy, Main memory, Auxiliary memory, Associative memory, Cache memories: concept and design issues, associative mapping, direct mapping, set-associative mapping, cache writing and initialization.	8
Total		42

Books:-

S.No.	Name of Books/Authors/Publisher
1)	Patterson, Computer Organization and Design, Elsevier Pub. 2009
2)	Morris Mano, Computer System Architecture, PHI 1992
3)	William Stalling, Computer Organization, PHI 2012
4)	Vravec, Hamacher&Zaky, Computer Organization, TMH 2009
5)	Tannenbaum, Structured Computer Organization, PHI 2006

Course Title	Course Structure			Pre- Requisite
Object-Oriented Software Engineering	L	T	P	Basics of Software Engineering
	3	0	2	

Course Objective:

The objective of object-oriented software engineering is to provide students with a comprehensive understanding of principles, practices, and techniques related to software development using object-oriented programming paradigms. It also focuses on applying the Unified Modeling Language (UML) for visualizing software designs, which includes use case diagrams, class diagrams, sequence diagrams, activity diagrams, etc. This subject also enables students to generate a detailed software requirement specification document. This subject aims to apply object-oriented principles to real-world problems, and case studies, demonstrating the relevance and applicability of object-oriented software engineering concepts.

Course Outcome (CO):

1. Understand the basics of software engineering, object-oriented paradigms, object-oriented methodologies used, and various basic terminologies.
2. Develop real-world software using conventional software development life cycle models, and object-oriented software development life cycle models.
3. Apply various techniques to gather requirements from the customers such as interviews, brainstorming session, FAST, and prototyping.
4. Design software requirement specification document, software design document, and test case matrix.
5. Design UML diagrams such as use case diagrams, class diagrams, sequence diagrams, state chart diagrams, and activity diagrams.
6. Analyze existing software by considering the issues of software risk management, and approaches to estimate the risk.

S.No.	Content	Contact Hours
Unit 1	Introduction: Object-oriented system concepts and Principles, Object Oriented Methodologies, Overview of Conventional Software Development Lifecycle Models, Fountain Model, Rational Unified Process, Some Basic Terminologies	8
Unit 2	Software Requirement Elicitation and Analysis: Software Requirement, Requirement Elicitation Techniques, Initial Requirement Documents, Use Case Approach, Characteristics of a Good Requirement, Software Requirements Specification Document, Requirement Change Management.	8
Unit 3	Object Oriented Analysis: Structured Analysis vs Object Oriented Analysis, Identification of Classes, Identification of Relationships,	9

	Identifying State and Behaviour.	
Unit 4	Object Oriented Design: What is Done in Object Oriented Design, Interaction Diagrams, Sequence Diagrams, Collaboration Diagrams, Refinement of Use Case Description, Refinement of Classes and Relationship, Identification of Operations to Reflect the Implementation Environment, Construction of Detailed Class Diagram, Development of Detailed Design, and Creation of Software Design Document, Generating Test Cases from Use Cases, Object Oriented Design, Principles for Improving Software Quality.	9
Unit 5	Moving Towards Implementation: Activity Diagram, Statechart Diagram, Storing Persistent data in Database, implementing the classes, and object-oriented testing. Real world case study/project based on object-oriented concepts and UML diagrams.	8
Total		42

Books:-

S.No.	Name of Books/Authors/Publisher
1	Y. Singh & R. Malhotra, "Object-Oriented Software Engineering", 1 st Ed., PHI Learning.
2	I. Jacobson, M. Christerson, P. Jonsson, G. Overgaard, "Object-Oriented Software Engineering", 2 nd Edition, Pearson Education.
3	I. Sommerville, "Software Engineering", Addison Wesley.

Course Title	Course Structure			Pre-Requisite
Machine Learning	L	T	P	Basic Programming
	3	0	2	

Course Objective:

The objective of the machine learning course is to understand the basic theory underlying machine learning. It will help students will be able to understand a range of machine learning algorithms along with their strengths, and weaknesses. Students will be able to apply machine learning algorithms to solve real-world problems to design optimized solutions and report on the expected accuracy that can be achieved by applying the models

Course Outcome (CO):

1. Understand the basic concepts of machine learning, supervised, unsupervised, regression analysis, and machine learning algorithms.
2. Apply the learned concepts of machine learning to interpret various problems.
3. Analyze the different mathematical machine learning models for various systems.
4. Evaluate the performance of the machine learning model using various performance measures.
5. Develop an efficient machine learning system to solve various real-time problems.

S.No.	Content	Contact Hours
Unit 1	Introduction to Machine Learning: Need, Objective, History of Machine Learning, Introduction of Machine Learning Approaches (Artificial Neural Network, Clustering, Reinforcement Learning, Classification, Regression), Types of Machine Learning Algorithms, Applications of Machine Learning, Data Science vs Machine Learning, Understanding the data-scale of measurement, Research Variables and Data analysis methods, Issues in Machine Learning Techniques, Steps in Model Prediction, Validation and performance evaluation, Confusion Matrix	8
Unit 2	Supervised Learning - I: Classification vs Regression. Decision Tree: Basic Methodology, ID3 Algorithm: Information Gain, Entropy, Inductive Bias, Occam's Razor, Issues in Decision Tree, Problem with IDE3 Algorithm, Problem with Information Gain Approach, C4.5 algorithm: Gain ratio and Decision Trees using Gain Ratio, CART algorithm: Gini Index and Decision trees using Gini Index, Implementing Decision trees, K-Nearest Neighbor- Introduction, Nearest Neighbor Algorithm, Feature Weighting in KNN, Random Forest, Ensemble Learning- Standard Ensemble Learning Strategies, Bagging, AdaBoost, Stacking.	8
Unit 3	Supervised Learning - II: Artificial Neural Network(ANN): Introduction, advantages and disadvantages Introduction, Characteristics of ANN, Topologies, Neuron and its	10

	<p>terminologies, ANN: Transfer Functions (Purelin, Sign, Step, Log Sigmoid, Tan Sigmoid, Unipolar and Bipolar), Perceptron, Perceptron Convergence Theorem.</p> <p>Single Layer ANN, Multilayer Perceptron: Back Propagation Learning Algorithm, Implementing Back propagation algorithm, ANN: Learning and Generalization, Bias and Variance, Bias/Variance Trade-off, Preventing Over-fitting and Under-fitting, Applications of Neural Networks.</p> <p>Support Vector Machine (SVM): Baic, Maximal Margin Hyperplane, Linear SVM, Non-Linear SVM, Attribute Transformation, Kernel Trick, Applications of SVM.</p> <p>Bayesian Learning: Probability Theory, Bayes theorem, Naïve Bayes Learning Algorithm, Nearest Neighbor Classifiers</p>	
Unit 4	<p>Unsupervised Learning: Introduction, Clustering Introduction and its applications, Partitional Clustering vs Hierarchical Clustering, Partial vs Complete Clustering, Hard Clustering vs Soft Clustering, K-means Clustering.</p> <p>Need and Classification of Dimensionality Reduction Methods, Univariate Analysis, Correlation-Based Feature Selection (CFS), Feature Extraction, and Principal Component Analysis-Variance, Covariance, Covariance Matrix, Eigenvectors, Eigen values, PCA</p>	8
Unit 5	<p>Recent Applications & Research Topics: Case Studies based on various Supervised Learning Approaches.</p> <p>Cross Validation Methods, Bias Variance trade-off, Techniques to deal with Imbalanced dataset, Resampling, Measuring, and Comparing two classification algorithms using statistical tests.</p>	8
Total		42

Books:-

S.No.	Name of Books/Authors/Publisher
1	Introduction to Machine Learning, Alpaydin, E., MIT Press, 2004
2	Machine Learning, Tom Mitchell, McGraw Hill, 1997.
3	Elements of Machine Learning, Pat Langley Morgan Kaufmann Publishers, Inc. 1995. ISBN 1-55860-301-8
4	The elements of statistical learning, Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. Vol. 1. Springer, Berlin: Springer series in statistics, 2001.
5	Machine Learning: A probabilistic approach, by David Barber, 2006.

Course Title	Course Structure			Pre-Requisite
Database Management Systems	L	T	P	MS Excel
	3	0	2	

Course Objective: To provide knowledge about the principles, concepts and applications of Database Management Systems.

Course Outcome (CO):

1. To understand the concepts of DBMS and would have acquired skills to analyse the real-world problem domains in the context of DBMS and demonstrate the same through ER diagram.
2. To apply and demonstrate with understanding of relational query languages such as SQL, Relational Algebra and Relational Calculus.
3. To relate the concepts of inference rules, data constraints and normalization. Students would also have acquired skills to identify application of the same.
4. Familiar with basic database storage structures and access techniques: file and page organizations, indexing methods including B tree, and hashing. To appraise the basic issues of Transaction processing and Serializability.
5. To classify various concurrency control techniques and recovery procedures
6. Familiar with case studies regarding commercial database, Oracle platforms, Postgres and MYSQL.

S.No.	Content	Contact Hours
Unit 1	Introduction: Database system concepts and its architecture, Data models schema and instances, Data independence and database language and interface, Data definition languages, DML. Overall database structure. Data modeling using Entity Relationship Model: E.R. model concept, notation for ER diagrams mapping constraints, Keys, Concept of super key, candidate key, primary key generalizations, Aggregation, reducing ER diagrams to tables, extended ER model.	7
Unit 2	Relational Data Model and Language: Relational data model concepts, integrity constraints, Keys domain constraints, referential integrity, assertions, triggers, foreign key relational algebra, relational calculus, domain and tuple calculus, SQL data definition queries and updates in SQL.	7
Unit 3	Data Base Design: Functional dependencies, normal forms, 1NF, 2NF, 3NF and BCNF, multi-valued dependencies fourth normal form, join dependencies and fifth normal form. Inclusion	6

	dependencies, lossless join decompositions, normalization using FD, MVD and JDs, alternatives approach to database design.	
Unit 4	File Organization, Indexing and Hashing Overview of file organization techniques, Indexing and Hashing- Basic concepts, Static Hashing, Dynamic Hashing, ordered indices, Multi-level indexes, B-Tree index files, B+- Tree index files, Buffer management Transaction processing concepts: Transaction processing system, schedule and recoverability, Testing of serializability, Serializability of schedules, conflict & view serializable schedule, recovery from transaction failures, deadlock handling.	8
Unit 5	Concurrency Control Techniques: Locking Techniques for concurrency control, time stamping protocols for concurrency control, concurrency control in distributed systems. multiple granularities and multi-version schemes. Case Studies: Commercial databases, Oracle, Postgress, MySQL	14
Total		42

Books:-

S.No.	Name of Books/Authors/Publisher
1	Elmasri, Navathe,"Fundamentals of Database systems", Addison Wesley, 2006.
2	Korth, Silbertz, Sudarshan,"Data base concepts", McGraw-Hill, 2001.
3	Ramakrishna, Gehkre, "Database Management System", McGraw Hill,2002.
4	Date C.J.,"An Introduction to Database systems", 2004

Course Title	Course Structure			Pre-Requisite
Algorithm Design and Analysis	L	T	P	Fundamentals of Programming
	3	1	0	

Course Objective:

To introduce the concept of algorithmic efficiency by analyzing various algorithms such as Searching, Sorting, Divide-and-Conquer algorithms and to know details about the Greedy Paradigm, Principle of Dynamic Programming, Back Tracking, Branch and Bound, and Computational Complexity.

Course Outcome (CO):

1. To learn the Algorithm and Design Concepts of linear and non-linear structures and complexity.
2. To understand the concept of searching and sorting
3. To learn concepts of searching and sorting.
4. To learn concepts of the Greedy method.
5. To understand concepts of Dynamic programming.
6. To understand the concepts of Branch and Bound.
7. To understand computational complexity.

S.No.	Content	Contact Hours
Unit 1	Introduction: Concept of algorithmic efficiency, run time analysis of algorithms, Asymptotic Notations. Growth of Functions, Master's Theorem.	6
Unit 2	Searching and Sorting: Structure of divide-and-conquer algorithms; examples: binary search, quick sort, Stassen Multiplication; merge sort, heap sort, and Analysis of divide and conquer run time recurrence relations.	7
Unit 3	Greedy Method: Overview of the greedy paradigm examples of exact optimization solution: minimum cost spanning tree, approximate solutions: Knapsack problem, Kruskal's algorithm and Prim's algorithm for finding Minimum cost Spanning Trees, Dijkstra's and Bellman Ford Algorithm for finding Single source shortest paths, Huffman coding, Activity Selection Problem.	8
Unit 4	Dynamic programming: Principles of dynamic	7

	programming. Applications: Rod cutting problem, Floyd-Warshall algorithm for all pair shortest paths. Matrix multiplication, Travelling salesman Problem, Longest Common sequence, Back tracking: Overview, 8-queen problem, and Knapsack problem, Traveling Salesman problem	
Unit 5	Branch and bound: LC searching Bounding, FIFO branch and bound, LC branch and bound application: 0/1 Knapsack problem. Computational Complexity: Complexity measures, Polynomial Vs non-polynomial time complexity; NP-hard and NP-complete classes, examples: Circuit Satisfiability, Vertex cover, Subset Sum problem, Randomized Algorithms, String Matching, NP-Hard and NP-Completeness, Approximation Algorithms, Sorting Network, Matrix Operations, Polynomials and FFT, Number Theoretic Algorithms.	14
Total		42

Books:-

S.No.	Name of Books/Authors/Publisher
1	T .H . Cormen, C . E . Leiserson, R. L. Rivest “Introduction to Algorithms”, 3 rd Ed., PHI.
2	E. Horowitz, S. Sahni, and S. Rajsekarani, “Fundamentals of Computer Algorithms,” Galgotia Publication, 2008.
3	Sara Basse, A. V. Gelder, “Computer Algorithms,” Addison Wesley, 1999.
4	Aho, Hopcroft, Ullman: The Design and Analysis of Algorithms, Addison Wesley, 1974.