

Sant Gadge Baba Amravati University, Amravati

Faculty of Science and Technology

Programme: M Sc Computer Science

PROGRAMME OUTCOMES (POs)

Upon completion of the programme successfully, students would be able to

PO1: Problem Analysis

Identify, formulate, review research literature and analyze complex engineering problems in Computer Science and Engineering reaching substantiated conclusions using first principles of mathematics, natural sciences and engineering sciences.

PO2: Design / Development of Solutions

Design solutions for complex engineering problems and design system components or processes of Computer Science and Engineering that meet the specified needs with appropriate consideration for public health and safety, cultural, societal and environmental considerations.

PO3: Conduct Investigations of Complex Problems

Use research-based knowledge and research methods including design of experiments in Computer Science and Engineering, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO4: Modern tool usage

Create, select and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex activities related to Computer Science with an understanding of the limitations.

PO5: The services to the society

Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice in Computer Science and Engineering.

PO6: Project Management

Demonstrate knowledge and understanding of the computer science and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PROGRAMME SPECIFIC OUTCOMES (PSOs)

Upon completion of the programme successfully, students would be able to

PSO 1: deliver efficient solutions for emerging challenges in the computation domain through continuous learning

PSO2

design, develop, implement computer programs and use knowledge in various domains to identify research gaps and hence to provide solution to new ideas and innovations.

Employability Potential in M.Sc. Computer Science

If you've studied computer science, you will have gained many technical and non-technical skills which are highly valued by employers, from leadership to programming. The increasing scope of computer science means you have plenty of choice in a wide variety of highly specialized areas.

Computer technologies are integral to modern life, so you're likely to find your computer science skills in high demand across many different industries. These include financial organizations, management consultancy firms, software houses, communications companies, data warehouses, multinational companies, governmental agencies, universities and hospitals.

As always, it's extremely beneficial to have completed relevant work experience. You should also consider compiling a portfolio of your own independent projects outside of your degree, which could be in the form of programming, moderating online or even building an app. This will demonstrate to employers your interest in the subject and your problem-solving skills, creativity and initiative.

- Application analyst.
- Applications developer.
- Cyber security analyst.
- Data analyst.
- Forensic computer analyst.
- Game designer.
- Games developer.
- Machine learning engineer
- Cyber security analyst
 - Data analyst
 - Forensic computer analyst
 - Game designer
 - Games developer
 - Machine learning engineer
 - Penetration tester
 - Software engineer
 - Systems analyst
 - UX designer
 - Web designer & Developer
 - Business analyst
- IT sales professional
- IT trainer
- Nanotechnologist
- Network Engineer
- Telecommunications researcher
- Database Manager/ Administrator

Common employers are IT consultancies and IT service providers. However, as most businesses rely on computers to function effectively, there are also opportunities within the IT departments of major organisations in sectors such as:

- Aerospace and Defence
- Agricultural
- Financial Services
- Healthcare
- Manufacturing
- Public And Third Sectors
- Telecommunications

- Banking
- E-Commerce
- Medical
- Defence
- Education
- Communication
- Automobile Industry
- Printing Industry
- Film Industry
- Entertainment Industry
- E- Governance
- Satellite Launching
- Simulators
- Research & Development
- And Lot More...

You can also find opportunities with a range of small to medium-sized enterprises (SMEs).

Another option is to set up your own business, providing IT services such as web design and consultancy.

Computing degrees combine theoretical study and practical projects, teaching you subject-specific skills including:

- Programming Languages
- Hardware Architecture And Construction
- Network Design and Engineering
- Software Engineering
- Multimedia Design
- Software Tools and Packages.

You'll learn how to specify, design and construct computer-based systems, evaluate and recognise potential risks and design creative solutions.

You'll also get more generic skills from your computing degree including:

- Teamwork and Leadership
- Communication
- Problem Solving
- Negotiation
- Time Management and Organisation
- Report Writing
- Numeracy
- Commercial Awareness.

Continuing professional development (CPD) is especially important when you're working with computers as technology and software develops at such a rapid pace.

**Prescribed Syllabus
Semester-I
Core Subjects**

Course Code	1MCSI	
Course Name	1 Computer System Design	
Total Credits	4	
Course Outcomes	<p>Course Outcome: Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. Understand the theory and architecture of central processing unit. 2. Analyze some of the design issues in terms of speed, technology, cost, performance. 3. Design a simple CPU with applying the theory concepts. 4. Use appropriate tools to design verify and test the CPU architecture. 5. Understand the theory of memory system design 6. Understand the architecture and functionality of central processing unit. 7. Exemplify in a better way the I/O and memory organization. 	
Units	Contents	Total Hrs
I	<p>Computer System Technology: From Components to Applications, Computer Systems and Their Parts, Generations of Progress, Processor and Memory Technologies, Peripherals, I/O, and Communications, Software Systems and Applications</p> <p>Computer Performance: Cost, Performance, and Cost/Performance, Defining Computer Performance, Performance Enhancement and Amdahl's Law, Performance Measurement vs Modelling, Reporting Computer Performance, The Quest for Higher Performance</p>	10
II	<p>Instructions and Addressing: Abstract View of Hardware, Instruction Formats, Simple Arithmetic and Logic Instructions, Load and Store Instructions, Jump and Branch Instructions, Addressing Modes</p> <p>Procedures and Data: Simple Procedure Calls, Using the Stack for Data Storage, Parameters and Results, Data Types, Arrays and Pointers, Additional Instruction</p> <p>Assembly Language Programming</p>	10
III	<p>Number Representation: Positional Number Systems, Digit Sets and Encodings, Number-Radix Conversion, Signed Integers, Fixed-Point Numbers, Floating-Point Numbers</p> <p>Adders and Simple ALUS: Simple Adders, Carry Propagation Networks, Counting and Incrementation, Design of Fast Adders, Logic and Shift Operations, Multifunction ALUS</p> <p>Multiplier and dividers, Floating-Point Arithmetic</p>	10
IV	<p>Instruction Execution Steps: A Small Set of Instructions, The Instruction Execution Unit, A Single-Cycle Data Path Branching and Jumping, Deriving the Control Signals, Performance of the Single-Cycle Design</p> <p>Control Unit Synthesis: A Multicycle Implementation, Clock Cycle and Control Signals, The Control State Machine, Performance of the Multicycle Design, Microprogramming, Dealing with Exceptions</p>	10
V	<p>Main Memory Concepts: Memory Structure and SRAM, DRAM and Refresh Cycles, Hitting the Memory Wall, Pipelined and Interleaved Memory, Non-volatile Memory, The Need for a Memory</p>	10

	<p>Hierarchy</p> <p>Cache Memory Organization: The Need for a Cache ,What Makes a Cache Work?, Direct-Mapped Cache , Set-Associative Cache , Cache and Main Memory, Improving Cache Performance</p> <p>Mass Memory Concepts ,Virtual Memory and Paging</p>	
VI	<p>Input/output Devices :</p> <p>Input/output Devices and Controllers ,Keyboard and Mouse ,Visual Display Units , Hard-Copy Input/output Devices ,Other Input/output Devices , Networking of Input/output Devices</p> <p>Input/output Programming : I/O Performance and Benchmarks ,Input/Output Addressing , Scheduled I/O: Polling, Demand-Based I/O: Interrupts , I/O Data Transfer and DMA ,Improving I/O Performance</p> <p> Buses, Links, and Interfacing, Context switching, interrupts</p>	10
	<p>Text Books:</p> <p>1)Computer Architecture From Microprocessors to supercomputers by Behrooz Parhami Oxford</p>	
	<p>Reference Books:</p> <ol style="list-style-type: none"> 1. Digital Logic and Computer Design Morris Mano, Pearson Education 2. Carl Hamacher, Z. Varnesic and S Zaky," Computer Organization". Tata McGraw-Hill, 5th Edition, ISBN: 13 9781259005275. 3. Stallings, William Computer organization and architecture designing for performance. Pearson Education India, 2000. 2. M. Morris, R. Mano Logic Fundamentals and Computer Design, Prentice Hall of India, 4 th Edition, 2007. 4. David A. Patterson and John L. Hannessey Computer Organization and Design: The Hardware and Software Interface, Morgan Kaufmann, Elsevier, 4th Edition, 2012. 5. John Hayes, Computer Architecture and Organization, Tata McGraw Hill, 5 th Edition, 1996.2002. 	

Course Code	IMCS2	
Course Name	2 Data Structure with OOP	
Total Credits	4	
Course Outcomes	<p>Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. Learn the concepts of linear data structures such as arrays, linked lists, stacks and queues. 2. Learn the concepts of non-linear data structures such as trees& Graph. 3. Learn and understand various data searching and sorting methods with its complexity. 4. Demonstrate operations such as insertion, deletion, searching and traversing on data structures. 5. analyse and apply specific sorting and searching methods depending upon factors like type of data, volume of data. 6. Learn & Understand B-tree indexing, hashing, collisions processing and its applications. 7. Learn the fundamental concepts of data structures. 8. Identify and Apply the fundamental concepts of data structures 9. Apply Academic Skills & Critical Thinking Skills 	
Unit	Content	Total Hrs

Unit I	Arrays, Matrices and Linked List, Arrays, Arrays the Abstract Data Type, Array Representation, Matrices, Special Matrices, The Linear List- Array Representation, Data Objects and Structures, The Linear List Data structure, Array Representation, Vector Representation, Multiple List in a Single Array, Linear Lists- Linked Representation, Singly Linked Lists and Chains, Circular	10
--------	--	----

	Lists and Header Nodes, Doubly Linked Lists, Sparse Matrices and its Representation, Searching in Array & Linked List.	
Unit II	Stacks and Queues, Stack: Definition and Application, Array Representation, Linked Representation, Applications, Queues: Definition and Application, Array Representation, Linked Representation, Applications, Priority Queue, Definition and Application.	10
Unit II	Trees: Terminology and Concepts, Binary and Other Trees, Trees, Binary Trees, Properties of Binary Trees, Representation of Binary Trees, Common Binary Trees Operations, Binary Trees Traversal, The ADT Binary Trees, The Class linked Binary Trees, Application, Binary Search Trees, Definition, Abstract Data Types, Binary Search Trees Operations and Implementation, Balanced Search Trees, AVL Trees.	10
Unit IV	Sorting: Internal Sorting, Optimal Sorting Time, Sorting Objects, Insertion Sort, Selection Sort, Merge Sort, Quick Sort, Heap Sort, Radix Sort, External Sorting: Run Generation, Sorting with Tapes, Sorting with Disks.(10	10
Unit V	Graphs: Definition, Representation of Graphs, Graph Implementation, Graph Traversals, Application of Graph Traversals, Minimum Cost Spanning Trees, Shortest Path Problems, All Pair Shortest Paths. (10	10
Unit VI	Indexing: Indexed Binary Search Trees, B-Trees: Indexed Sequential Access Method (ISAM), m-Way Search Trees, B-Trees of Order m, Height of a B- Tree, Searching a B-Tree, Inserting into a B-Tree, Deletion from a B-Tree, Node Structure, Hashing: Hash Function, Collision Resolution, Rehashing, Extensible Hashing.(10	10
	Text books: 1. "Data structure algorithms and Applications in C++" : Sartaj Sahani (Second Edition Universities Press)	
	Reference Books: 1. "Introduction to Data Structures" - Bhagat Singh & T.L. Naps. 2. "Data structures using C"-Tanenbaum, Langsam, Augenstein PHI 3. "Classic Data Structures", - D. Samanta PHI 4. "Data structure and Program design in C" - Kruse, Leung, Tondo (PHI) 5. "Data structure" - Tenenbaum 6. "Data structure and algorithm analysis in C++ ": Mark Allan Welss, Addison Wesley	

Course Code	1MCS3
Course Name	3 Data Base Management Technologies
Total Credits	4

Course Outcomes	Course Outcomes: On completion of this course, students would be able to: CO1: Analyze Database Management System & its Architecture and design ER model. CO2: Implement database queries using database languages and normalize the database design using normal forms. CO3: Write queries and PL/SQL Code blocks for given requirements, using different SQL and PL/SQL concepts. CO4: Apply different query processing, optimizing, indexing and hashing techniques in real-time database. CO5: Apply Transaction Management concepts, concurrency control concepts and	
------------------------	---	--

	<p>deadlock handling concepts in real-time situations</p> <p>CO6: Use advanced database Programming concepts like Parallel databases, Distributed database & Object-based database for processing data.</p>	
Unit	Contents	Total Hrs
Unit I	<p>Introduction to Database System and Data Modelling: Introduction to Database Management Systems, Purpose of Database Systems, Database-System Applications, View of Data, Data Models, Database Languages, Relational Databases, Database Architecture/ Structure.</p> <p>Database Design and Data Modelling using ER model & EER model : Entity & its types, Attributes, Relationships, Constraints, Keys, Design Process, Entity Relationship Model, ER Diagram, Design Issues, Extended E-R Features, Converting E-R & Enhanced -ER diagram into tables.</p> <p>Case Study: Design ER Model for any real time application and convert the same into tables on paper.</p>	10
Unit II	<p>Relational Database Design: Introduction to Relational Data Model: Structure of Relational Databases, Database Schema, Keys, Schema Diagrams, Relational Query Languages, Relational Operations, Attributes and Domains, CODD's Rules.</p> <p>Relational Integrity Constraints: Key Constraint, Referential Integrity constraint, Entity Integrity constraint, Domain constraint.</p> <p>Relational Database Design: Features of Good Relational Designs, Normalization, Atomic Domains and First Normal Form, Decomposition using Functional Dependencies, Algorithms for Decomposition, 2NF, 3NF, BCNF.</p> <p>Case Study: Convert ER-Diagram of Unit 1 to Relational Database and apply Normalization.</p>	10
Unit III	<p>SQL and PL/SQL: SQL: Characteristics and Advantages, SQL Data Types and Literals, DDL, DML, DCL, TCL, SQL Operators. Tables: Creating, Modifying, Deleting, Updating. SQL DML Queries: SELECT Query and clauses, Index and Sequence in SQL. Views: Creating, Dropping, Updating using Indexes, Set Operations, Predicates and Joins, Set membership, Tuple Variables, Set comparison, Ordering of Tuples, Aggregate Functions, SQL Functions, Nested Queries. Relational Algebra and Relational Calculus. PL/SQL: Concept of Stored Procedures and Functions, Cursors, Triggers, Assertions, Roles and Privileges.</p> <p>Case Study: Implementation of unit 1 case study using SQL/PLSQL.</p>	10
Unit IV	<p>Storage and File Structure: Overview of Physical Storage, Magnetic Disk and Flash Storage, RAID, Tertiary Storage, File Organization, Organization of Records in Files, Data-Dictionary Storage, Database Buffer,</p> <p>Indexing and Hashing: Ordered Indices, B+-Tree Index Files and its Extensions, Static Hashing and Dynamic Hashing, Comparison of Ordered Indexing and Hashing, Bitmap Indices, Some General Issues Concerning Indexing.</p> <p>Query Processing and Optimization: Measures of Query Cost, Query Operation: Selection, Sorting and Join Operation, Evaluation of Expression, Transformation of Relational Expressions, Estimating Statistics of Expression Results, Choice of Evaluation Plans.</p>	10

Unit V	<p>Transaction Management: Transaction concept, Transaction states, ACID properties, Transaction and Schedules, Concurrent Execution of Transactions, Lock Based Concurrency Control .</p> <p>Concurrency control and Recovery System: ,Serializability and Recoverability, Lock management, Specialized locking techniques, Concurrency control without locking, Dealing Deadlocks: Deadlock handling, detection, prevention & Recovery.</p> <p>Recovery System: Shadow-Paging Recovery and Log-Based Recovery, Checkpointing, Crash recovery. Database Security: Database Security Issues, Discretionary Access Control Based on Grant & Revoking Privilege, Mandatory Access Control and Role Based Access Control for multilevel security.</p>	10
Unit VI	<p>Advances in Databases Emerging Databases:</p> <p>Parallel Database: I/O Parallelism, Interquery Parallelism ,Intraquery Parallelism, Intraoperation Parallelism, Interoperation Parallelism, Query Optimization.</p> <p>Distributed Databases: Homogeneous and Heterogeneous Databases, Distributed Data Storage, Distributed Transactions, Commit Protocols, Concurrency Control in Distributed Databases, Availability, Distributed Query Processing, Heterogeneous Distributed Databases, Cloud-Based Databases.</p> <p>Object-Based Databases: Complex Data Types, Structured Types and Inheritance in SQL, Table Inheritance, Array and Multiset Types in SQL, Object-Identity and Reference Types in SQL , Implementing O-R Features, Object-Relational Mapping.</p> <p>Case Study-Use of NoSQL/MapReduce/Hadoop/MongoDB databases for processing unstructured data from social media with Nested Data Types like : JSON/ XML/ Object-Relational Database System/ Spatial Data</p>	10
	<p>Text books:</p> <ol style="list-style-type: none"> 1. “Database System Concepts”, 6th Edition by Abraham Silberschatz, Henry F.Korth, S. Sudarshan, McGraw-Hill. 	
	<p>Reference Books:</p> <ol style="list-style-type: none"> 1. S.K.Singh, “Database Systems : Concepts, Design and Application”. 2. C. J. Date, "An Introduction to Database Systems", Addison-Wesley, 8th Edition 3. Connally T, Begg C., "Database Systems- A Practical Approach to Design, Implementation and Management", Pearson Education, 5th Edition. 4. Elmasri and S B Navathe,“Fundamentals of Database Systems” 7th edition, Pearson. 5. Ivan Bayross, "SQL, PL/SQL the Programming Language of Oracle", BPB Publications. 6. Kevin Roebuck, "Storing and Managing Big Data - NoSQL, HADOOP and More". 	

Course Code	1MCS4
Course Name	4 Computer Network & Wireless Communications
Total Credits	4
Course Outcomes	<p>Course Outcome:</p> <p>Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. Apply cellular concepts to evaluate the signal reception performance in a cellular network and traffic analysis to design cellular network with given quality of service constraints. 2. Determine the type and appropriate model of wireless fading channel based on the system parameters and the property of the

	<p>wireless medium.</p> <ol style="list-style-type: none"> 3. Analyse and design receiver and transmitter diversity techniques. 4. Determine the appropriate transceiver design of multi-antenna systems and evaluate the data rate performance. 5. Design wireless communication systems with key 3G (e.g., CDMA) and 4G (OFDM) technologies. 6. Describe and differentiate four generations of wireless standard for cellular networks. 	
Units	Contents	Hrs
I	<p>Introduction to Wireless Communication Systems: Examples of Wireless Communication Systems: Paging Systems, Cordless Telephone Systems, Cellular Telephone Systems How a Cellular Telephone Call is Made, Comparison of Common Wireless Communication Systems, Trends in Cellular Radio and Personal Communications.</p> <p>modern wireless communication system: Wines Local Loop (WIL) and LMDS, Wireless Local Area Networks (WLAN) ,Bluetooth and Personal Area Networks (PANs)</p> <p>The Cellular Concept-System Design Fundamentals: Introduction Frequency Reuse, Channel Assignment Strategies ,Handoff Strategies ,Practical Handoff Considerations</p> <p>Mobile Radio Propagation: Large-Scale Path Loss : Introduction to Radio Wave Propagation, Free Space Propagation Model ,Relating Power to Electric Field The Three Basic Propagation Mechanisms</p> <p>Mobile Radio Propagation: Small-Scale Fading and Multipath : Small- Scale Multipath Propagation, Factors Influencing Small-Scale Fading , Doppler Shift Impulse Response Model of a Multipath Channel, Relationship Between Bandwidth and Received Power, Small-Scale Multipath Measurements , Direct RF Pulse System, Spread Spectrum Sliding Correlator Channel Sounding Frequency Domain Channel Sounding</p>	10
II	<p>Modulation Techniques for Mobile Radio: frequency modulation vs. Amplitude modulation.</p> <p>Equalization ,Diversity, and Channel Coding : Introduction, Fundamentals of Equalization, Training A Generic Adaptive Equalizer , Algorithms for Adaptive Equalization, Zero Forcing Algorithm, Least Mean Square Algorithm, Recursive Least Squares Algorithm ,Summary of Algorithms, Fractionally Spaced Equalizers , Diversity Techniques, Derivation of Selection Diversity Improvement, Derivation of Maximal Ratio Combining Improvement, Practical Space Diversity Considerations, Selection Diversity, Feedback or Scanning Diversity, Maximal Ratio Combining, Equal Gain Combining, Polarization Diversity, Frequency Diversity, Time Diversity, Interleaving, Fundamentals of Channel Coding, Convolutional Codes, Decoding of Convolutional Codes, The Viterbi Algorithm ,Other Decoding Algorithms for Convolutional Codes),Coding Gain .</p>	10

III	<p>Speech Coding :Introduction, Characteristics of Speech Signals, Quantization Techniques , Uniform Quantization, Non-uniform Quantization, Adaptive Quantization , Vector Quantization, Adaptive Differential Pulse Code Modulation (ADPCM) , Frequency Domain Coding of Speech, Sub-band Coding, Adaptive Transform Coding, Vocoders, Channel Vocoders, Formant Vocoders, Cepstrum Vocoders , Voice-Excited Vocoder, Linear Predictive Coders, LPC Vocoders, Multipulse Excited LPC, Code-Excited LPC, Residual Excited LPC, Choosing Speech Codes for Mobile Communications, The GSM Codec, The USDC Codec, Performance Evaluation of Speech Coders.</p>	10
IV	<p>Multiple Access Techniques for Wireless Communications: Introduction, Introduction to Multiple Access , Frequency Division Multiple Access (FDMA), Time Division Multiple Access (TDMA) , Spread Spectrum Multiple Access, Frequency Hopped Multiple Access (FHMA), Code Division Multiple Access(CDMA), Hybrid Spread Spectrum Techniques, Space Division Multiple Access(SDMA) , Packet Radio, Packet Radio Protocols , Pure ALOHA, Slotted ALOHA, Carrier Sense Multiple Access (CSMA) Protocols, Reservation Protocols, Reservation ALOHA, Packet Reservation Multiple Access (PRMA),Capture Effect in Packet Radio, Capacity of Cellular Systems, Capacity of Cellular CDMA, Capacity of CDMA with Multiple Cells, Capacity of Space Division Multiple Access.</p>	10
V	<p>Wireless Networking : Introduction to Wireless Networks, Differences Between Wireless and Fixed Telephone Networks, The Public Switched Telephone Network (PSTN), Limitations in Wireless Networking, Merging Wireless Networks and the PSTN , Development of Wireless Networks, First Generation Wireless Networks, Second Generation Wireless Networks ,Third Generation Wireless Networks, Fixed Network Transmission Hierarchy , Traffic Routing in Wireless Networks, Circuit Switching, Packet Switching, The X.25 Protocol, Wireless Data Services, Cellular Digital Packet Data (CDPD), Advanced Radio Data Information Systems (ARDIS), RAM Mobile Data (RMD),Common Channel Signaling (CCS),The Distributed Central Switching Office for CCS, Integrated Services Digital Network (ISDN), Broadband ISDN and ATM, Signaling System No. 7 (SS7), Network Services Part (NSP) of SS7, Message Transfer Part (MTP) of SS7, Signaling Connection Control Part (SCCP) of SS7, The SS7 User Part, Integrated Services Digital Network User Part (ISUP), Transaction Capabilities Application Part (TCAP),Operation Maintenance and Administration Part (OMAP), Signaling Traffic in SS7,SS7 Services , Performance of SS7,An Example of SS7-Global Cellular Network Interoperability, Personal Communication Services/Networks (PCS/PCNs) , Packet vs. Circuit Switching for PCN, Cellular Packet-Switched Architecture Network Functionality in Cellular Packet-Switched Architecture, Protocols for Network Access, Packet Reservation Multiple Access (PRMA), Network Databases, Distributed Database for Mobility Management, Universal Mobile Telecommunication System (UMTS)</p>	10
VI	<p>Wireless Systems and Standards: AMPS and ETACS, AMPS and ETACS System Overview, Call Handling in AMPS and ETACS, AMPS and ETACS Air</p>	10

	<p>Interface, N-AMPS ,United States Digital Cellular 11.2.1 USDC Radio Interface (IS-54 and IS-136), United States Digital Cellular Derivatives (IS-94 and IS-136) ,Global System for Mobile (GSM),GSM Services and Features, GSM System Architecture, GSM Radio Subsystem, GSM Channel Types, GSM Traffic Channels (TCHS) , GSM Control Channels (CCH),Example of a GSM Call, Frame Structure for GSM, Signal Processing in GSM, CDMA Digital Cellular Standard (IS-95),Frequency and Channel Specifications ,Forward CDMA Channel, Convolutional Encoder and Repetition Circuit, Block Interleaver, Long PN Sequence ,Data Scrambler Power Control Subchannel, Orthogonal Covering, Quadrature Modulation Reverse CDMA Channel, Convolutional Encoder , Block Interleaver Orthogonal Modulation and Symbol Repetition, Variable Data Rate Transmission, Direct Sequence Spreading ,Quadrature Modulation, IS-95 with 14.4 kbps Speech ,CT2 Standard for Cordless Telephones Coder [ANS95],CT2 Services and Features , The CT2 Standard, Digital European Cordless Telephone (DECT) ,Features and Characteristics, DECT Architecture, DECT Functional Concept, DECT Radio Link, PACS Personal Access Communication Systems ,PACS System Architecture, PACS Radio Interface, Pacific Digital Cellular (PDC), Personal Handy phone System (PHS), US PCS and ISM Bands, US Wireless Cable Television, Summary of Standards Throughout the World.</p>	
	<p>Text Books: Theodore S. Rappaport : Wireless Communication (II Edition), Pearson</p>	
	<p>Reference Books: 1. Vern A.D.”Wireless Data Technologies” : Wiley 2. Ray ES : “Space/Terrestrial Mobile Network Internet access & QOS, :Wiley 3. Rai Zing : “Multihop Wireless Networks, Wiley 4. Yu Kwang, Ricky Klark, VincentK.N.Lau : wireless Internet & Mobile Computing Interoperability & Performance” : Wiley</p>	

Skill-I

Course Code	1MCS5
Course Name	4-Advanced Java/ Ns2/ Tools
Credits	
Course Outcomes	<p>1. Advanced Java: Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. become familiar with the advanced features of Java Language 2. Design a desktop application which can used for many kind of clients. 3. Gain the knowledge of J2EE architecture, MVC Architecture. 4. Design a web application which can work as a dynamic web with the help of JDBC.

	<ol style="list-style-type: none"> 5. Develop an application which can also be connected with the database. 6. To understand Java Servlets and their life cycle 7. To understand Java server Pages (JSP) technology 8. To develop JSP Custom tags and use them in JSP pages <p>2. NS-2 Tool: Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. demonstrate operation of network 2. simulate and demonstrate the performance of GSM and CDMA 3. implement data link layer and transport layer protocols. 4. Demonstrate Installation procedure of the required software in groups and document the same in the journal.
--	--

Sr. No.	Contents	Total Hrs
	The following information can be used as guidelines for basic understanding of Advance Java Programming and NS-2 tool.	
	<p>Advanced Java: It is an advanced technology or advance version of Java specially designed to develop web-based, network-centric or enterprise applications. It includes the concepts like Servlet, JSP, JDBC, RMI, Socket programming, etc. It is a specialization in specific domain. Most of the applications developed using advance Java uses tow-tier architecture i.e. Client and Server. All the applications that run on Server can be considered as advance Java applications.</p> <p>JEE (advance Java) provides libraries to understand the concept of Client-Server architecture for web- based applications. It is also important understand the advance Java if you are dealing with trading technologies like Hadoop, cloud-native and data science. There is a number of advance Java frameworks like, Spring, Hibernate, Struts, that enables us to develop secure transaction-based web applications such as banking application, inventory management application.</p> <p>Topic covers: Basics of a Web application, Web Container and Web Application Project Set up, J2EE and Web Development, Advance Networking, , Java Server Pages, Hibernate Framework , Java Web Frameworks: Spring MVC, Spring Boot Framework, Java Server Faces, Session Management, JSP Tag library</p> <p>Examples:</p>	

	<ol style="list-style-type: none"> 1) Write a client program to send any string from its standard input to the server program. The server program reads the string, finds number of characters and digits and sends it back to client program. Use connection-less communication. 2) Write down Five Basic steps to establish JDBC connection from Java Application. Also mention sample code for each step. 3) Write down the Program for testing the Servlet and study deployment descriptor. 4) Write down the Program which displays the simple JSP file. 5) Create database of student subject-wise data and retrieve all data using JSP and generate xml structure along with DTD and XML Schema definition 6) Using Spring Template manages Database and Transaction. <p>Use JSF Standard Components and Facelets Tags.</p>	
	<p>NS-2 tool: NS2 stands for Network Simulator Version 2. It is an open-source event-driven simulator designed specifically for research in computer communication networks.</p> <p>Features of NS2:</p> <ol style="list-style-type: none"> 1. It is a discrete event simulator for networking research. 2. It provides substantial support to simulate bunch of protocols like TCP,FTP, UDP, https and DSR. 3. It simulates wired and wireless network. 4. It is primarily Unix based. 5. Uses TCL as its scripting language. 6. Otcl: Object oriented support 7. Tccl: C++ and otcl linkage 8. Discrete event scheduler <p>Basic Architecture:</p> <p>NS2 consists of two key languages: C++ and Object-oriented Tool Command Language (OTcl). While the C++ defines the internal mechanism (i.e., a backend) of the simulation objects, the OTcl sets up simulation by assembling and configuring the objects as well as scheduling discrete events. The C++ and the OTcl are linked together using TcCL</p>	

	<p>Following steps illustrate how to install the packages separately</p> <ol style="list-style-type: none"> 1. All the files will be zip format 2. Now let's begin with installing Tcl. 3. Install Tk 4. Install OTcl: 5. Install Tclcl-1.19 6. Install ns-2.34: 7. Install NAM 8. Install xgraph <p style="text-align: center;">Examples:</p> <ol style="list-style-type: none"> 1) To create scenario and study the performance of token bus protocol through NS2 simulator. 2) To create scenario and study the performance of token ring protocols through NS2 simulator. 3) To Simulate and to study stop and Wait protocol by using NS-2 Simulator. 4) To simulate and study the Distance Vector routing algorithm by using NS-2 Simulator. 5) To simulate and study the link state routing algorithm by using NS-2 Simulator. 6) To create scenario and study the performance of CSMA / CD protocol through NS-2 Simulator. 7) To Simulate and to study of Go Back N protocol through NS-2 Simulator. 8) To simulate a wireless sensor network using NS2. 9) To simulate a Mobile Adhoc network (MANET) using NS2. 10) To implement a Transport Control Protocol in sensor network through the Network Simulator -2 	
--	---	--

Elective-I

Course Code	1MCS6(1)	
Course Name	Discrete Mathematical Structure	
Credits	4	

Course Outcomes	<p>Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. discuss fundamental concepts and tools in discrete mathematics 2. Understand sets and perform operations and algebra on sets. Determine properties of relations, identify equivalence and partial order relations, sketch relations. Identify functions and determine their properties. 3. assess the curricular skills acquired by students at college level through Assignments, Unit test, Internal Test, Group Discussion/Seminar/Mini Project, Study Tour 4. Discuss fundamental concepts and tools in discrete mathematics with emphasis on their applications to computer science. 5. Use mathematically correct terminology and notation. 6. Construct correct direct and indirect proofs. 7. Apply logical reasoning to solve a variety of problems. 	
Unit	Content	Total hours
Unit I	Mathematical logic: Introduction, statements and notations, connectives – negation, conjunction, disjunction, Statement formulas and truth tables, conditional, bi- conditional, well formed formulas, Tautologies, Equivalence of formulas, Duality law, Tautological implications, functionally complete sets of connectives, other connectives, Normal and principal normal forms, completely parenthesized infix and polish notations, Theory of inference for statement calculus – validity using truth table, rules of inference, consistency of premises and indirect method of proof.	10
Unit II	Set theory: Basic concepts of set theory, representation of discrete structures, relations and ordering: relations, properties of binary relations in a set, relation matrix and graph of a relation, partition and covering of a set, equivalence relation, compatibility relations, composition of binary relations, Functions – composition of functions, Inverse function.	10
Unit III	Algebraic Structures: Algebraic systems: Examples and general properties, Semigroups and monoids, Grammar and Languages, Polish expressions and their compilation, Groups Definition and examples, subgroups and homomorphism, cosets and Lagrange’s theorem, Group codes – the communication model and basic notions, generation of codes by using parity checks, error recovery in group codes	10
Unit IV	Lattices and Boolean algebra: Lattice as POSETs, definition, examples and properties, Lattice as algebraic systems, sublattices, Direct product and homomorphism, Special lattices, Boolean algebra - definition and examples, subalgebra, Direct product and homomorphism, Boolean functions, representation and minimization of Boolean Finite state machines.	10
Unit V	Graph theory: Basic concepts of graph theory – definitions, paths, reachability and connectedness, matrix representation, Storage representation and manipulation of graphs- trees, representation and operations, list structures and graphs, Simple precedence grammars-syntax terminology, a view of parsing, notion and use of precedence relations, formal definition of precedence relations.	10
Unit VI	Fault detection in combinational switching circuits – Faults in combinational circuits, Notions of Fault detection, Algorithm for generating a fault matrix, procedure for detection of faults; Introduction to computability theory: Finite-state acceptors and regular grammars, Turing machines and partial recursive functions	10

	<p>Text books:</p> <ol style="list-style-type: none"> 1. Discrete Mathematical Structures – Bernard Kolman, Robert Busby, S.C. Ross and Nadeemur- Rehman (Pearson Education) 	
	<p>Reference Books:</p> <ol style="list-style-type: none"> 1. Discrete Mathematical Structures with applications to computer science- J. P. Tremblay & R.. Manohar (McGraw Hill Editions) 2. Discrete mathematics - Semyour Lipschutz, Marc Lipson (MGH), Schaum’s outlines. 3. Discrete mathematics and its applications - Kenneth H. Rosen (AT&T Bell Labs)(mhhe.com/ rosen) 	
	<p>Course Material/Learning Resources</p> <p>Weblink to Equivalent MOOC on SWAYAM if relevant:</p> <p>https://onlinecourses.nptel.ac.in/noc20_cs37/preview</p> <p>https://onlinecourses.nptel.ac.in/noc19_cs67/preview</p> <p>https://onlinecourses.swayam2.ac.in/cec20_ma02/preview</p> <p>https://www.classcentral.com/course/swayam-discrete-mathematics-5217</p> <p>Weblink to Equivalent Virtual Lab if relevant:</p> <p>https://virtuallabs.merlot.org/vl_math.html</p> <p>http://vlabs.iitb.ac.in/vlabs-dev/labs/mit_bootcamp/signals/labs/exp1/simulation.php</p> <p>https://computing.llnl.gov/topic/discrete-mathematics</p> <p>Any pertinent media (recorded lectures, YouTube, etc.) if relevant:</p> <p>https://www.youtube.com/watch?v=wGLTV8MgLIA&list=PLU6SgdYc_YsfJ27O0dvmwafS3X8CecqUg</p> <p>https://www.youtube.com/watch?v=YBb2oYIzXK0&list=PLxCzCOWd7_ah2wwES9vPWsEL6ipTaUSl3</p> <p>https://www.youtube.com/watch?v=NuGDkmwEObM</p> <p>https://www.youtube.com/watch?v=pCyuJgmr7W8 https://www.youtube.com/watch?v=q4L-wUF3yig</p>	

Course Code	1MCS6(2)
Course Name	Entrepreneurship Development
Total Credits	04

Course Outcomes	Course Outcome: Upon completion of this course successfully, students would be able to <ul style="list-style-type: none"> ○ understand the Key concepts underpinning entrepreneurship and its application ○ analyse the recognition and popularization of product/ service/ process opportunities. 	
	<ul style="list-style-type: none"> ○ apply new ideas, methods and ways of thinking ○ create new business plan and marketing it. ○ investigate the issues associated with securing and managing financial resources in new and established organisations. ○ design creative strategies for pursuing, exploiting and further developing new opportunities. 	
Units	Contents	Total Lectures
I	<p>The Nature and Importance of Entrepreneurs: Nature and Development of Entrepreneurship, Definition of Entrepreneur today, The Entrepreneurial Decision Process, Types of Startups, Role of Entrepreneurship in Economic Development, Entrepreneurial Careers and Education, Ethics and Social Responsibility of Entrepreneurs, The Future of Entrepreneurship.</p> <p>The Entrepreneurial and Intrapreneurial Mind: The Entrepreneurial Process, Managerial versus Entrepreneurial Decision Making, Causes for Interest in Intrapreneurship, Corporate versus Intrapreneurial Culture, Climate for Intrapreneurship, Intrapreneurial Leadership Characteristics, Establishing Intrapreneurship in the Organization, Problems and Successful Efforts.</p>	10
II	<p>The Individual Entrepreneur: Entrepreneurial Feelings, Entrepreneur Background and Characteristics, Motivation, Role Models and Support Systems, Male versus Female Entrepreneurs, Minority Entrepreneurship, Entrepreneurs versus Inventors</p> <p>International Entrepreneurship Opportunities: The Nature of International Entrepreneurship, The Importance of International Business to the Firm, International versus Domestic Entrepreneurship, Entrepreneurial Entry into International Business, Barriers to International Trade</p>	10
III	<p>Creativity and the Business Idea: Sources of New Ideas, Methods of Generating Ideas, Creative Problem Solving, Opportunity Recognition, Product Planning and Development Process, E-Commerce and Business Start-Up</p> <p>Legal Issues for Entrepreneur: Intellectual Property, Need for a Lawyer, Selecting a Lawyer, Legal Issues in Setting Up the Organization, Patents, Business Method Patents, Trademarks, Copyrights, Trade Secrets, Licensing, Product Safety and Liability, Insurance, Contracts</p>	10
IV	<p>The Business Plan: Creating And Starting The Venture: Planning as Part of the Business Operation, Write the Plan, Scope and Value of the Business Plan-Read the Plan, Potential Lenders and Investors Evaluate the Plan, Presenting the Plan, Information Needs, Using the Internet as a Resource Tool, Writing the Business Plan, Using and Implementing the Business Plan, Investigating the failure of Business Plan.</p> <p>The Marketing Plan: Industry Analysis, Marketing Research for the New Venture, Understanding the Marketing Plan, Characteristics of a Marketing Plan, The Marketing Mix, Steps in Preparing the Marketing Plan, Contingency Planning, Why Some Plans Fail</p>	10

V	<p>The Organizational Plan: Developing the Management Team, Legal Forms of Business, Tax Attributes of Forms of Business, The Limited Liability Company versus the S Corporation, S Corporation, The Limited Liability Company, Designing the Organization, Building the Management Team and a Successful Organization Culture, The Role of a Board of Directors, The Board of Advisors ,The Organization and Use of Advisors</p> <p>The Financial Plan: Operating and Capital Budgets, Pro Forma Income Statement, Pro Forma Cash Flow, Pro Forma Balance Sheet , Break-Even Analysis , Pro Forma Sources and Applications of Funds , Software Packages</p>	10
VI	<p>Sources Of Capital: An Overview, Personal Funds, Family and Friends, Commercial Banks, Role of SBA in Small Business Financing, Research and Development Limited Partnerships, Government Grants, Private Placement, Bootstrap Financing</p> <p>Informal Risk Capital and Venture Capital: Financing the Business, Informal Risk-Capital Market, Venture Capital, Valuing Your Company, Deal Structure</p>	10
	<p>Text Books: Robert D. Hisrich, Michael P. Peters, Dean A. Shepherd: “Entrepreneurship” (Sixth Edition) Tata McGraw Hill Education Private Limited, Special Indian Edition.</p>	
	<p>Reference Books:</p> <ol style="list-style-type: none"> i. Rajeev Roy “Entrepreneurship” Oxford HigherEducation. ii. Colombo Plan Staff College for Technical Education, Manila “Entrepreneurship Development” Tata McGraw-Hill. iii. Vasant Desai “Entrepreneurship Development” Himalaya Publishing House. 	

Course Code	1MCS6(3)	
Course Name	Research Methodology	
Total Credits	4	
Course Outcomes	<p>Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. draw upon foundational knowledge, learn, adapt and successfully apply analytical and computational approaches on changing societal and technological challenges 2. Demonstrate the ability to choose methods appropriate to research aims and objectives. 3. Understand the limitations of particular research methods. <p>The curricular skills acquired by students should be assessed at college/university department level through Assignments,Unit test, Internal Test, Group Discussion/Seminar/Mini Project, Study Tour</p>	
Unit	Content	Total Hrs

Unit I	Introduction: Meaning of Research, Objectives of Research, Motivation in Research, Types of Research, Research Approaches, Significance of Research, Research Methods versus Methodology, Research and Scientific Method, Importance of Knowing How Research is Done, Research Process, Criteria of Good Research	10
Unit II	Defining the Research Problem: What is a Research Problem, Selecting the Problem, Necessity of Defining the Problem, Technique Involved in Defining a Problem Research Design: Meaning of Research Design, Need for Research Design, Features of a Good Design, Important Concepts Relating to Research Design, Different Research Designs, Basic Principles of Experimental Designs	10
Unit III	Methods of Data Collection: Collection of Primary Data, Observation Method, Interview Method, Collection of Data through Questionnaires, Collection of Data through Schedules, Difference between Questionnaires and Schedules, Some Other Methods of Data Collection, Collection of Secondary Data, Selection of Appropriate Method for Data Collection	10
Unit IV	Processing and Analysis of Data: Processing Operations, Some Problems in Processing, Elements/Types of Analysis, Statistics in Research, Measures of Central Tendency, Measures of Dispersion, Measures of Asymmetry (Skewness), Measures of Relationship, Simple Regression Analysis, Multiple Correlation and Regression, Partial Correlation, Association in Case of Attributes, Other Measures	10
Unit V	Sampling Design: Census and Sample Survey, Implications of a Sample Design, Steps in Sampling Design, Criteria of Selecting a Sampling Procedure, Characteristics of a Good Sample Design, Different Types of Sample Designs, Select a Random Sample, Random Sample from an Infinite Universe, Complex Random Sampling Designs	10
Unit VI	Interpretation of Data and Paper Writing: Layout of a Research Paper, Journals in Computer Science, Impact factor of Journals, When and where to publish ? Ethical issues related to publishing, Plagiarism and Self-Plagiarism.	10
	Text books: 1. Handbook of Research Methodology- Dr. Shanti Bhushan Mishra, Dr. Shashi Alok Educreation Publishing	
	Reference Books: 1. Business Research Methods – Donald Cooper & Pamela Schindler, TMGH, 9th edition 2. Business Research Methods – Alan Bryman & Emma Bell, Oxford University Press. 3. Research Methodology – C.R. Kothari	

Course Code	1MCS6(4)
Course Name	Management Information System
Total Credits	04

Course Outcomes	Course Outcome: Upon completion of this course successfully, students would be able to <ol style="list-style-type: none">1. Understand the leadership role of Management Information Systems in achieving business competitive advantage through informed decision making.2. Analyze and synthesize business information and systems to facilitate evaluation of strategic alternatives.

	<p>3. Effectively communicate strategic alternatives to facilitate decision making. Evaluate the role of information systems in today's competitive business environment.</p> <p>4. Assess the relationship between the digital firm, electronic commerce, electronic business and internet technology.</p>	
Units	Contents	Total Lectures
I	MIS concepts, definition, Role, Impact of MIS, MIS and computers, MIS and academics, MIS support to Management, Role and importance of management. MIS and process of management MIS in organization structure and strategic management business.	10
II	Basics of MIS: Decision making, Decision methods, behavioural concepts, organizational decision making, MIS and decision make concepts, Information; concepts and classification, Methods of data and information collection: value of information, organization and information. Humans an information processor. Development of MIS and choice of IT.	10
III	Applications of MIS: Applications in manufacturing sector, applications in service sector, Introduction to service, sector, Creating a destructive services, MIS applications in service industries and role of MIS in source industries. DSS: Concepts and philosophy, deterministic systems and knowledge based expert systems. MIS and role of DSS. MIS in Enterprise Management System.	10
IV	Technology in MIS: Data processing, Transaction processing, Application processing, Information System processing, TQM of IS. DBMS: Object Oriented Technologies, client Server Arch. And MIS.	10
V	MIS and Networks: Network Topology, LAN, Data Communication, ATM Technology, Business Process Reengineering: Introduction BP, Process Model of organization, Value stream model, Delays in BP, Relevance of IT, MIS and BPR.	10
VI	MIS and Data warehouse: Architecture, Design and Justification of data warehouse, organization. Management and implementation of data -warehousing, E-Business: Models, WWW, E-payment, security in E-business, MIS and E-business.	10
	<p>Text Books: W. S. Jawadekar : Management Information System (II Edition), (TMH)</p>	
	<p>Reference Books: 1. Kenneth C. Landon & J. P. Landon.: Management Information System, 8th Ed. Pearson Education. 2. Voichdan, Homer, Information Systems for operation & Mgmt. 3. A. K.Gupta and J.K. Sharma: Management of Systems (Macmillan)</p>	

Course Code	1MCS6(5)
Course Name	Data Science and Analytics
Total Credits	04
	Upon completion of this course successfully, students would be able to
Course Outcomes	1. Analyze and interpret data using an ethically responsible

	<p>approach</p> <ol style="list-style-type: none"> 2. Use appropriate models of analysis, assess the quality of input, derive insight from results, and investigate potential issues 3. Apply computing theory, languages, and algorithms, as well as mathematical and statistical models, and the principles of optimization to appropriately formulate and use data analyses 4. Formulate and use appropriate models of data analysis to solve hidden solutions to business-related challenges 5. Interpret data findings effectively to any audience, orally, visually, and in written formats 	
Units	Contents	Total Lectures
I	The Art of Data Science: Volume, Velocity, Variety, Machine Learning , Supervised and Unsupervised Learning, Predictions and Forecasts , Innovation and Experimentation, The Dark Side- Big Errors, Privacy, Theories, Models, Intuition, Causality, Prediction, Correlation	10
II	Review of Maths : Exponentials, Logarithms, and Compounding , Normal Distribution ,Poisson Distribution, Moments of a continuous random variable, Combining random variables , Vector Algebra Statistical Regression , Diversification , Matrix Calculus , Matrix Equations	10
III	Open Source: Modelling in R: System Commands , Loading Data, Matrices, Descriptive Statistics, Higher-Order Moments, Quick Introduction to Brownian Motions with R , Estimation using maximum- likelihood, GARCH/ARCH Models ,Introduction to Monte Carlo, Portfolio Computations in R , Finding the Optimal Portfolio, Root Solving, Regression, Heteroskedasticity, Auto-regressive models, Vector Auto-Regression , Logit , Probit ,Solving Non-Linear Equations, Web- Enabling R Functions	10
IV	MoRe: Data Handling and Other Useful Things: Data Extraction of stocks using quantmod , Using the merge function , Using the apply classof functions , Getting interest rate data from FRED ,Cross-Sectional Data (an example) , Handling dates with lubridate ,Using the data.table package , Another data set: Bay Area Bike Share data ,Using the plyr package family	10
V	Being Mean with Variance: Markowitz Optimization: Quadratic (Markowitz) Problem, Solving the problem with the quadprog package, Tracing out the Efficient Frontier, Covariances of frontier portfolios: rp,rq , Combinations, Zero Covariance Portfolio ,Portfolio Problems with Riskless Assets, Risk Budgeting	10

VI	Learning from Experience: Bayes Theorem: Introduction, Bayes and Joint Probability Distributions, Correlated default (conditional default), Continuous and More Formal Exposition , Bayes Nets , Bayes Rule in Marketing ,Other Applications	10
	Text Books: Data Science : Theories, Models, Algorithms and Analytics by Sanjiv Ranjan Das	
	Reference Books: I. Beginning Data Science in R by Thomas Mailund , Apress Publications II. Getting Started with Data Science, making sense of data with analytics by Murtaza Haider, Published by Pearson plc	

Laboratories

Course Code	1MCS7	
Course Name	Lab-I 1,2 - Programming(C/C++/Java/ALP)	
Total Credits	--	
Course Outcomes	<p>Course Outcome: Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> Demonstrate the use of various OOPs concepts with the help of programs. Describe the procedural and object oriented paradigm with concepts of classes, functions, data and objects. Demonstrate the programs for the implementation of constructors, destructors and function overloading. Use the syntax and semantics of java programming language. Design event driven GUI and web related applications which mimic the real word scenarios. Get Familiarized with the assembly level programming. 	
	Contents	Total Hours
	<p>The sample list of programs is given below. This list can be used as a guideline for problem statements but the scope of the laboratory should not be limited to the same. Aim of the list is to inform about minimum expected outcomes.</p> <ol style="list-style-type: none"> Write a program in C to perform various operations to implement Arrays. Write a program in C to perform various arithmetic operations for implementing Pointers. Write a program in C to implement Structure Programming. Write a program in C to implement various File Handling operations. Write a program in C to implement various loops and a control statement. Write a program in C++ to implement Class and Object. Write a program in C++ to implement function overloading. Write a program in C++ to implement constructor and destructor. Write a program in C++ to implement abstract base classes. Write a program in C++ to implement Inline function. Write a program in JAVA that connects to a database using JDBC. 	

- | | | |
|--|---|--|
| | <ol style="list-style-type: none">12. Write a program in JAVA for handling Mouse events and Key events.13. Write a program in JAVA that implements a multi-thread application.14. Write a program in JAVA that simulates a traffic light using GUI.15. Write a program in JAVA that computes factorial value using Applet.16. Write a program in JAVA to implement Concept of matrices.17. Write a program in JAVA to create user defined package.18. Programs in JAVA to implement Exception Handling.19. Write an ALP program to implement reverse of a given string.20. Write a program in ALP for addition of two 16-bit number using 16-bit registers.21. Write a program in ALP to exchange of two 8-bit number using XCHG instructions.22. Write a program in ALP to increment and decrement of two 8-bit number23. Write an ALP program to find the LCM & HCF of given numbers.24. Write an ALP program to move a block of data from one memory location to the other.25. Write an ALP program to perform multi byte addition and subtraction. | |
|--|---|--|

GIC

Course Code	1MCS8	
Course Name	Lab-II 3-SQL/ DBMS tools, MS-SQL, My Sql	
Total Credits	--	
Course Outcomes	<p>Course Outcomes: Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. Demonstrate the Basics Concepts and SQL Queries of Database Management System. 2. Apply the Conceptual Design Model and Database Hierarchical Structure to construct the real-world requirement. 3. Analyze the various constraints to populate the database through SQL Queries. 4. Implement different working concepts of DBMS using SQL Queries. 5. Present the result of database creation and querying process, document it. 	
	Contents	Total Hours
	<p>The sample list of programs is given below. This list can be used as a guideline for problem statements but the scope of the laboratory should not be limited to the same. Aim of the list is to inform about minimum expected outcomes.</p> <ol style="list-style-type: none"> 1. Implementation of Creating and managing SQL table. 2. Concept Design with E-R Model 3. Different data types and its implementation 4. Implementation of Relational Operators 5. Implementation of Logical operator 6. Implementation of Aggregate function 7. Implementation of Constraints Not null, Unique, Primary key, Foreign key, Check, Default, Create index 8. Implementation of DDL commands 9. Implementation of DML and DCL commands 10. Implementation of TCL commands 11. Perform Join ,Inner join, Outer join, Natural join 12. Implementation of Advance queries, Subquery and Grouping 13. Implementation of Cursors 14. Implementation of Triggers 15. Implementation of Procedures and functions 16. Implementation of Views 17. Creating Database /Table Space <ul style="list-style-type: none"> • Managing Users: Create User, Delete User • Managing roles:-Grant, Revoke. 	

Course Name		GIC1: User Experience Design
Unit	Contents	
Unit I	What is UX Strategy?, The Four Tenets of UX Strategy, Validating the Value Proposition, Conducting Competitive Research.	
Unit II	Conducting Competitive Analysis, Storyboarding Value Innovation, Creating Prototypes for Experiments.	
Unit III	Conducting Guerrilla User Research, Designing for Conversion, Strategists in the Wild	
Reference Book: UX Strategy by Jaime Levy, Foreword by Jason Calacanis, SPD ORILLY Publication.		

Course Name		GIC2: Effective Email Communication
Units	Contents	
Unit I	Why Email: <ul style="list-style-type: none"> • Email: The Hearth of the Internet, • Email's Undeserved Bad Rap, • The different Types of Email Communication: Email Newsletters, Catalog Emails, Announcement Emails, Press Releases, Sales and Sign-up Process Emails. 	
Unit II	Planning and Email Campaign: <ul style="list-style-type: none"> • Planning is Essential, • Meeting Our Client: The Modern Henschman Magazine Client Briefing, • Setting Goals, • Measuring Success, • Planning Your Content: An Email: An Email is Not a Website, Email in the Real World, Planning the Modern Henschman Newsletter. 	
Unit III	Design for the Inbox: <ul style="list-style-type: none"> • Does Email really need Designing?, • Designing Plain Text Email: Guidelines for a Readable Plain Text Email, • The Plain Text Version of the Modern Henschman Newsletter, • The Case for HTML Email, • Designing HTML Email: The Design Environment for Email, Essential Elements of an Effective Email, Adapting a Website Design in to an Email Design, Layout Possibilities, Designing to Meet Business Goals. 	
Reference Book: Creating Stunning HTML Email by Mathew Patterson, sitepoint SPD publication.		

**Semester-II
Core Subjects**

Course Code	2MCS1
Course	1 Operating System Algorithms

Name		
Total Credits	4	
Course Outcomes	<p>Course Outcome: Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. Analyze basic components of Operating Systems and various types of Operating Systems. 2. understand and implement the mechanism of an OS to handle multiple processes and threads and their communication. 3. analyze different mechanisms involved in memory management OS. 4. find the loopholes in the system to resolve the problem of deadlock. 5. study the components and management aspects of concurrency management 6. study and analyze the problems in distributed Operating System. 	
Units	Contents	Total Lectures
I	<p>Design Techniques I : Design Techniques, Interface Design-Overview, Motivation, Applicability, Consequences, Related Design Techniques, Connection in Protocols- Overview, Motivation, Operating System Examples, Computer Science Examples, Applicability, Consequences, Implementation issues and variations, related design techniques.</p> <p>Implementing Processes:- The system call interface, Implementation of Processes:- Process Creation, Process States, Process Dispatching, The System Stack, Timer Interrupts, System call interrupt handling- copying messages between address spaces, Implementation of Waiting- Waiting for Messages, Waiting inside a system Call, suspending System Calls</p> <p>Parallel Systems- Parallel Hardware, An Operating System for a Two-Processor, Race Conditions with a Shared Process Table, Atomic Actions, A Multiprocessor Operating System, The current process variable, dispatching with a shared process table, busy waiting, handling the queues, grouping of shared variables, a general solution, using two process tables.</p>	10
II	<p>Interprocess Communication patterns- Patterns of Interprocess communication- Competing and Cooperating</p> <p>Processes:-Everyday Scheduling, First-Come. First-Served Scheduling ,Shortest Job- First Scheduling, highest-Response-Ratio-Next Scheduling, Priority Scheduling, Deadline Scheduling, Round-Robin Scheduling, Summary, Preemptive Scheduling Methods - Scheduling Overview, Round-Robin Scheduling, Heavily Loaded Systems, Two Queues, Multiple Queues, Two phase locking, starvation, Message Passing Variations- Using PIDs as Message, Message passing with non blocking sends, remote procedure calls, Semaphores:- Specifications of Semaphores operations, Implementation of Semaphore, an analogy, mutual exclusion with Semaphores, Rendezvous with Semaphores, procedure- consumer with Semaphores, counting Semaphores, procedure consumer with Semaphores, Semaphores and messages</p>	10
III	<p>Design Techniques II:- Indirection- overview, motivation , operating system examples, computer science example, discussion, applicability, consequences, Using State Machines - overview, operating system examples, computer science example, applicability, consequences, implementation issues and variations, Win Big, Then Give Some Back - overview, operating system examples, computer science example,</p>	10

	<p>applicability, consequences, Using Models for Inspiration- overview, operating system examples, computer science example, applicability, consequences</p> <p>Memory Management- Levels of Memory Management, Linking and loading a process-creating a load module, loading a load module, allocating memory in a running process, Dynamic memory allocation, Examples of dynamic memory allocation- logical and physical memory, allocating memory to processes, static memory management, handling variable sized processes, Multiprogramming Issues, Memory protection, memory management system calls- static allocation of memory to processes, dynamic of memory to processes, What about New and Malloc, Freeing Memory at Each Level, A Different Memory Management system call</p>	
IV	<p>Virtual Memory - Sharing the processor and sharing the money, swapping-efficient resources use and user needs, Overlays- Overlays in PC, Implementing Virtual Memory- hardware required to support virtual memory, software required to support virtual memory, What is the cost of Virtual Memory- Paging more than one process, locality, virtual memory management, Daemons and Events, File Mapping- the system call interface, an example of using file mapping, advantages of file mapping, memory and file mapping on the IBM 801</p> <p>Virtual Memory Systems - Global Page replacement algorithms - measuring the performance of page replacement algorithm, optimal page replacement, theories of program paging behavior, random page replacement, First In first out FIFO Page, Least Recently Used Page Replacement, Approximations of LRU, Clock Algorithm, Paging the operating system address space- locking page in memory, page sizes- reason for large page size, reason for small page size, clustering pages, Segmentation- What is segment, virtual memory with segmentation, segmentation with paging, history of segmentation, segmentation terminology</p>	10
V	<p>Design Techniques III :- Multiplexing - overview, operating system examples, computer science example, applicability, consequences</p> <p>I/O Devices:- I/O system software- Device driver, device driver interface, the two categories of device drivers, the block device interface, the character device interface, Device numbers, Unification of files and I/O devices,</p> <p>Generalised disk device drivers- Partitioning large disk, combining disks into a large logical disk, RAM disk, Memory as a device, Pseudo-ttys, Disk Caching</p> <p>File System:- File system Organization - What is a file system, File system structure, the file system descriptor, variations in file system layout, file system in disk partitions, combining file system, network mounting of file systems</p>	10
VI	<p>Design Techniques IV - Caching- overview, motivation , operating system examples, computer science example, discussion, applicability, consequences, implementation issues and variations, related design techniques,</p> <p>Resource Management - Integrated Scheduling, Queuing models of Scheduling, Real time operating system, protection of resources- users and processes, the importance of protection of resources, resources that needs protection, what we are protecting against, authorization, authentication, security and protection analogies, general strategy for protection, parts of protection system, User Authentication - Password, system architecture, other methods of authentication, password variations, identifying objects, identifying as a person</p> <p>Client - Server Model - Three modes of communication, system processes- Overview,</p>	10

	the initial process, system constant, initialization, interrupt handling, handling system calls, the system calls handling code, user knowledge of message queue identifiers, protection of resources, disk interrupt handler, Disk I/O system process, Server Data Structure, Micro kernel Operating system - Tradeoffs of the client server model, Object oriented operating systems	
	Text Books: 1. Operating System - A design oriented approach Charles Crowley Tata McGraw Hill Edition	
	Reference Books: 1. Modern Operating Systems, Pearson Education A. S. Tanenbaum 2. Advanced Concepts in Operating System M. Singhal & N. Shivaratri 3. Distributed Operating System by A. S. Tanenbaum 4. Linux Kernel, 2nd Edition By Daniel P. Bovet, Oreilly 5. The Design of Unix Operating System Maurice Bach, Pearson	

Course Code	2MCS2	
Course Name	2 Graphics Application programming	
Total Credits	4	
Course Outcomes	<p>Course Outcome: Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. Learn the fundamental concepts of Computer Graphics. 2. Identify and Apply the fundamental concepts of Computer Graphics in Animation, Virtual Reality. 3. Apply Academic Skills & Critical Thinking Skills 4. understand the mathematical modelling of graphical objects to be drawn/used in different kind of applications. 5. Learn and understand the concepts of computer graphics, including viewing, projection, perspective, modelling and transformation in 2D & 3D. 6. Learn and understand the algorithms to generate line segments, polygon and its transformations, windowing and clipping. 7. Demonstrate operations such as various Transformation and Projection. 8. Demonstrate various algorithms for scan conversion and filling of basic objects and their comparative analysis. 9. Implement display control, 3D geometry, primitives and conversions, algorithms for hidden surfaces and lines, concepts of shading and curves. 	
Units	Contents	Total Lectures
Unit I	Geometry and line generation: Introduction, Points and Lines, Planes and Coordinates, Line segments, Perpendicular line segments, Vectors, Pixels and Frame Buffers, Vector generation, Character generation, Displaying the frame buffer. Graphics primitive: Introduction, Display devices, Primitive operations, The Display-File Interpreter, Normalized	10

	device coordinates, Display-file structure, Display control.	
Unit II	Polygon: Introduction, Polygon , Polygon representation, Entering polygon, An inside test, Filling polygon, Antialiasing. Transformations: Introduction, matrices, scaling transformations, sin and cos, sum of angles, identifiers, rotation, homogeneous coordinates and translation, rotation about an arbitrary point, other transformations, display procedures.	10
Unit II	Segments: Introduction, the segment table, segment creation, closing a segment, deleting a segment, renaming a segment, visibility, image transformations, saving and showing segments, other display file structures, Some Raster techniques, Windowing and clipping: Introduction, viewing transformation, implementation, clipping, clipping the polygon, Adding Cclipping to the system, Generalized Clipping.	10
Unit IV	Interaction : Introduction, hardware, input devices-handling Algorithm, Event handling, Sample devices, The delectability attributes, Simulating a Locator with a Pick and Pick with a Locator, Echoing, Interactive Techniques. Three dimensions: Introduction, 3D Geometry, Primitives and Transformations, Rotation about an arbitrary axis, Parallel projection, Perspective projection, Viewing parameters, Conversion to View Plane Coordinates, The 3D Viewing Transformation.	10
Unit V	Hidden Surfaces and Lines: Introduction, Back face removal & algorithm, Z Buffers, Scan-Line algorithm, The Painter's algorithm, Comparison Techniques, Warnock's algorithm, Franklin algorithm, Hidden Linemethod, Binary Space Partition, An Application.	10
Unit VI	Shading: Introduction, diffusion, illumination, point source illumination, specular reflection, transparency and shadows. Curves: Introduction, curve generation, implementation, interpolating polygons, E-splines, B-Splines and Curves.	10
	Text books: 1. "Computer Graphics A Programming approach"- Steven Harington PHI	
	Reference Books: 1. "Computer Graphics, C Version" - Donald Hearn, M. Pauline Baker - 2. (Prentice Hall Press) 3. "Interactive Computer Graphics"- Newmann and Sproul 4. "Computer Graphics"- Rogers.	

Course Code	2MCS3	
Course Name	3 Software Engineering	
Total Credits	4	
Course Outcomes	<p>Course Outcome: Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. identify the need for engineering approach to software development and various processes of requirements analysis for software engineering problems. 2. Analyse various software engineering models and apply methods for design and development of software projects. 3. Work with various techniques, metrics and strategies for Testing software projects. 4. Identify and apply the principles, processes and main knowledge areas for Software Project Management 5. Proficiently apply standards, CASE tools and techniques for engineering software projects 	
Units	Contents	Total Lectures
I	<p>The Software Process: Prescriptive Process Models: The Waterfall Model, Incremental Process Models, Evolutionary Process Models, Concurrent Models, A Final Word on Evolutionary Processes. Agile Development: What is an Agile Process?, Agility Principles, The Politics of Agile Development, Human Factors. Extreme Programming (XP): XP Values, The XP Process, Industrial XP, The XP Debate. Requirements Modeling : Scenarios, Information And Analysis Classes: Requirements Analysis: Overall Objectives and Philosophy, Analysis Rules of Thumb, Domain Analysis, Requirements Modeling Approaches, Scenario-Based Modeling: Creating a Preliminary Use Case, Refining a Preliminary Use Case, Writing a Formal Use Case, Design Concepts: Design within the Context of Software Engineering, The Design Process: Software Quality Guidelines and Attributes, The Evolution of Software Design, Design Concepts: Abstraction, Architecture, Patterns, Separation of Concerns, Modularity, Information Hiding, Functional Independence, Refinement, Aspects, Refactoring, Object-Oriented Design Concepts, Design Classes, Architectural Design: Software Architecture: What is Architecture?, Why Is Architecture Important?, Architectural Descriptions, Architectural Decisions, Architectural Genres, Architectural Styles: A Brief Taxonomy of Architectural Styles, Architectural Patterns, Organization and Refinement, Architectural Design: Representing the System In Context, Defining Archetypes, Refining the Architecture Into Components, Describing Instantiations of the System, Component- Level Design: What Is a Component? An Object-Oriented View, The Traditional View, A Process Related View, Designing Class-Based Components, Basic Design Principles, Component-Level Design Guidelines, Cohesion, Coupling, User Interface Design: The Golden Rules: Place the User in Control, Reduce the User's Memory Load, Make the Interface Consistent, User Interface Analysis and Design: Interface Analysis and Design Models, The Process, Pattern-Based Design: Design Patterns, Kinds of Patterns, Frameworks, Describing a Pattern, Pattern Languages and Repositories, WebApp Design: WebApp Design Quality, Quality Management: Quality Concepts: What Is Quality?, Software Quality, Garvin's Quality Dimensions, McCall's Quality Factors, Targeted</p>	10

	Quality Factors, The Transition to a Quantitative View	
II	<p>Software Quality Assurance: Elements of Software Quality Assurance, SQA Tasks, Goals and Metrics: SQA Tasks, Goals, Attributes, and Metrics, Formal Approaches to SQA: Statistical Software Quality Assurance: A Generic Example, Six Sigma for Software Engineering, Software Reliability: Measures of Reliability and Availability, Software Safety, Software Testing Strategies: A Strategic Approach to Software Testing: Verification and Validation, Organizing for Software Testing, Software Testing Strategy The Big Picture, Criteria for Completion of Testing, Strategic Issues, Test Strategies for Conventional Software: Unit Testing, Integration Testing, Test Strategies for Object-Oriented Software: Unit Testing in the OO Context, Integration Testing in the OO Context, Testing Conventional Applications: Software Testing Fundamentals, Internal and External Views of Testing, White Box Testing, Basis Path Testing: Flow Graph Notation, Independent Program Paths, Deriving Test Cases, Graph Matrices, Control Structure Testing: Condition Testing, Data Flow Testing, Loop Testing, Black Box Testing: Graph Based Testing Methods, Equivalence Partitioning, Boundary Value Analysis, Orthogonal Array Testing, Model Based Testing</p>	10
III	<p>Testing Web Applications: Testing Concepts for WebApps: Dimensions of Quality, Errors within a WebApp Environment, Testing Strategy, Test Planning, The Testing Process-An Overview, Content Testing: Content Testing Objectives, Database Testing, User Interface Testing: Interface Testing Strategy, Testing Interface Mechanisms, Testing Interface Semantics, Usability Tests, Compatibility Tests, Component-Level Testing, Navigation Testing: Testing Navigation Syntax, Testing Navigation Semantics, Formal Modeling And Verification: The Cleanroom Strategy, Functional Specification: Black Box Specification, State Box Specification, Clear Box Specification, Cleanroom Design: Design Refinement, Design Verification, Cleanroom Testing: Statistical Use Testing, Certification, Software Configuration Management: An SCM Scenario, Elements of a Configuration Management System, Baselines, Software Configuration Items, The SCM Repository: The Role of the Repository, General Features and Content, SCM Features, The SCM Process: Identification of Objects in the Software Configuration, Version Control, Change Control, Configuration Audit, Status Reporting, Configuration Management for WebApps: Dominant Issues, WebApp Configuration Objects, Content Management, Change Management, Version Control, Auditing and Reporting</p>	10
IV	<p>Product Metrics: A Framework for Product Metrics: Measures, Metrics, and Indicators, The Challenge of Product Metrics, Measurement Principles, Goal-Oriented Software Measurement, The Attributes of Effective Software Metrics, Metrics for the Requirements Model: Function-Based Metrics, Metrics for Specification Quality, Metrics for the Design Model: Architectural Design Metrics, Metrics for Object-Oriented Design, Class-Oriented Metrics-The CK Metrics Suite, Class-Oriented Metrics-The MOOD Metrics Suite, OO Metrics Proposed by Lorenz and Kidd, Component Level Design Metrics, Operation- Oriented Metrics, User Interface Design Metrics, Managing Software Projects:Project Management Concepts: The Management Spectrum: The People, The Product, The Process, The Project, The Product: Software Scope, Problem Decomposition, The Process: Melding the Product and the Process, Process Decomposition, Process And</p>	10

	<p>Project Metrics: Metrics in the Process and Project Domains: Process Metrics and Software Process Improvement, Project Metrics, Software Measurement: Size-Oriented Metrics, Function-Oriented Metrics, Reconciling LOC and FP Metrics, Object-Oriented Metrics, Use Case-Oriented Metrics, WebApp Project Metrics, Metrics for Software Quality: Measuring Quality, Defect Removal Efficiency</p>	
V	<p>Estimation For Software Projects: Observations on Estimation, The Project Planning Process, Software Scope and Feasibility, Resources: Human Resources, Reusable Software Resources, Environmental Resources, Software Project Estimation, Decomposition Techniques: Software Sizing, Problem-Based Estimation, An Example of LOC-Based Estimation, An Example of FP- Based Estimation, Process-Based Estimation, An Example of Process-Based Estimation, Estimation with Use Cases, An Example of Use Case-Based Estimation, Reconciling Estimates, Project Scheduling: Basic Concepts, Project Scheduling: Basic Principles, The Relationship Between People and Effort, Effort Distribution, Defining a Task Set for the Software Project: A Task Set Example, Refinement of Software Engineering Actions, Scheduling: Time-line Charts, Tracking the Schedule, Tracking Progress for an OO Project, Scheduling for WebApp Projects, Risk Management: Reactive versus Proactive Risk Strategies, Software Risks, Risk Identification: Assessing Overall Project Risk, Components and Drivers, Risk Projection: Developing a Risk Table, Assessing Risk Impact, Risk Refinement</p>	10
VI	<p>Maintenance And Reengineering: Software Maintenance, Software Supportability, Reengineering, Business Process Reengineering, Business Processes, A BPR Model, Software Reengineering: A Software Reengineering Process Model, Software Reengineering Activities, Reverse Engineering: Reverse Engineering to Understand Data, Reverse Engineering to Understand Processing, Reverse Engineering User Interfaces, Restructuring: Code Restructuring, Data Restructuring, Forward Engineering: The Economics of Reengineering Software Process Improvement: What Is SPI?: Approaches to SPI, Maturity Models, Is SPI for Everyone?, The SPI Process: Assessment and Gap Analysis, Education and Training, Selection and Justification, Installation/Migration, Evaluation, Risk Management for SPI, Critical Success Factors, The CMMI, The People CMM, Other SPI Frameworks, SPI Return on investment, SPI Trends</p>	10
	<p>Text Books: 1. Software Engineering: A Practitioner’s Approach. Roger S. Pressman, Seventh edition McGraw Hill International Edition.</p>	
	<p>Reference Books: 1. Software Engineering, A Precise Approach, Pankaj Jalote, Wiley India, 2010. 2. Software Engineering: A Primer, Waman S Jawadekar, Tata McGraw-Hill, 2008 3. Fundamentals of Software Engineering, Rajib Mall, PHI, 2005 4. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press. 5. Software Engineering1: Abstraction and modeling, Diner Bjorner, Springer Internationaledition, 2006. 6. Software Engineering2: Specification of systems and languages, Diner Bjorner,</p>	

<p>Springer International edition 2006.</p> <p>7. Software Engineering Foundations, Yingxu Wang, Auerbach Publications, 2008.</p> <p>8. Software Engineering Principles and Practice, Hans Van Vliet, 3rd edition, John Wiley & Sons Ltd.</p> <p>10. Software Engineering 3: Domains, Requirements, and Software Design, D. Bjorner, Springer International Edition.</p> <p>10. Introduction to Software Engineering, R. J. Leach, CRC Press.</p>	
---	--

Course Code	2MCS4	
Course Name	4 Data Mining and Data Warehousing	
Total Credits	04	
Course Outcomes	<p>Course Outcome: Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. use basic concepts and techniques of Data Mining 2. develop skills of using recent data mining software for solving practical problems. 3. gain experience of doing independent study and research. 4. study the methodology of engineering legacy databases for data warehousing and data mining to derive business rules for decision support systems. 5. Develop and apply critical thinking, problem-solving, and decision-making skills. 	
Units	Contents	Total Lectures
I	Introduction, Data Mining Functionalities, Data Preprocessing: Data Cleaning, Data Integration and Transformation, Data Reduction, Data Discretization and Concept Hierarchy Generation.	10
II	Data Warehouse and OLAP Technology: Overview, A Multidimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, From Data Warehousing to Data Mining. Data Cube Computation and Data Generalization: Efficient Methods for Data Cube Computation, Data Generalization and Concept Description.	10
III	Mining Frequent Patterns, Associations, and Correlations: Basic Concepts, Efficient and Scalable Frequent Itemset Mining Methods, Mining Various Kinds of Association Rules, From Association Mining to Correlation Analysis, Constraint-Based Association Mining.	10
IV	Classification and Prediction: Issues, Classification by Decision Tree Induction, Bayesian Classification, Rule-Based Classification, Classification by Backpropagation. Prediction: Linear Regression, Nonlinear Regression, Accuracy and Error Measures, Evaluating the Accuracy of a Classifier or Predictor.	10

V	Cluster Analysis: Partitioning Methods, Hierarchical Methods, Density-Based Methods, Grid-Based Methods, Model-Based Clustering Methods, Clustering High-Dimensional Data. Mining Time-Series Data, Mining Sequence Patterns in Biological Data.	10
VI	Graph Mining, Social Network Analysis and Multirelational Data Mining. Mining Object, Spatial, Multimedia, Text, and Web Data, Data Mining Applications, Trends in Data Mining.	10
	Text Books: Data Mining: Concepts and Techniques - J. Han, M. Kamber	
	Reference Books: 1.. Data Mining: Introductory and Advanced Topics – Margaret H.Dunham, Pearson Education 2. Data Warehousing in the real world - Sam Anahory, Dennis Murry, Pearson Education 3. Principles of Data Mining - David Hand, Heikki Manila, Padhraic Symth, PHI 4. Data Warehousing, Data Mining & OLAP, Alex Bezon, Stephen J. SmithMcGraw-Hill Edition 5. Data Warehousing Fundamentals, Paulraj Ponniah, Wiley-Interscience Publication	

SKILL-II

Course Code	2MCS5
Course Name	1-Operating system (Windows/Android/Linux)
Credits	4
Course Outcomes	<p>1. Operating system Algorithm: Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. implement the mechanism of an OS to handle multiple processes and threads and their communication. 2. simulate and demonstrate the performance mechanisms involved in memory management OS. 3. demonstrate the loopholes in the system to resolve the problem of deadlock. 4. simulate and demonstrate the performance the components and management aspects of concurrency management 5. simulate and demonstrate the performance of the problems in distributed Operating Systems. <p>2. Java: Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. become familiar with the advanced features of Java

	<p>Language</p> <ol style="list-style-type: none"> 2. Design a desktop application which can be used for many kinds of clients. 3. Gain the knowledge of J2EE architecture, MVC Architecture. 4. Design a web application which can work as a dynamic web with the help of JDBC. 5. Develop an application which can also be connected with the database. 6. apply Java Servlets and their life cycle 7. use Java server Pages (JSP) technology 8. develop JSP Custom tags and use them in JSP pages 	
S r. No.	Contents	Total Hrs
	The following information can be used as guidelines for basic understanding of Operating system Algorithms using Java.	
	<p>UNIX: It is a multi-user operating system. Developed at AT & T Bell Industries, USA in 110610. Ken Thomson and Dennis Ritchie developed it from the MULTICS (Multiplexed Information and Computing Service) OS. By 11080, UNIX had been completely rewritten using C language.</p> <p>LINUX: It is similar to UNIX, which is created by Linus Toruualds. All UNIX commands work in Linux. Linux is open source software. The main feature of Linux is coexisting with other OS such as Windows and UNIX.</p> <p>STRUCTURE OF A LINUX SYSTEM: It consists of three parts. a) UNIX kernel b) Shells c) Tools and Applications UNIX KERNEL: Kernel is the core of the UNIX OS. It controls all tasks, schedules all Processes and carries out all the functions of the OS. Decides when one program tops and another starts.</p> <p>SHELL: Shell is the command interpreter in the UNIX OS. It accepts command from the user and analyses and interprets them</p> <p>Topics Cover:-FCFS, SJF, Priority, Round robin CPU Scheduling algorithm, resource allocation, producer – consumer problem using semaphores, First fit/Best Fit / Worst Fit memory allocation algorithms, FIFO, LRU and Optimal Page Replacement algorithm, FCFS Disk Scheduling algorithm, Deadlock detection algorithm, SCAN and CSCAN Disk Scheduling algorithm, system calls of the operating system.</p> <ol style="list-style-type: none"> 1. Write a program to demonstrate the use of FCFS CPU Scheduling algorithm. 2. Write a program to demonstrate the use of SJF CPU Scheduling algorithm. 3. Write a program to demonstrate the use of Priority CPU Scheduling algorithm. 4. Write a program to demonstrate the use of Round Robin CPU Scheduling algorithm. 	

	<ol style="list-style-type: none"> 5. Write a program to demonstrate the use of resource allocation to identify whether the system is in safe state. 6. Write a program to implement the producer – consumer problem using semaphores. 7. Write a program to demonstrate the use of First fit/Best Fit / Worst Fit memory allocation algorithms. 8. Write a program to demonstrate the use of FIFO Page Replacement algorithm. 9. Write a program to demonstrate the use of LRU Page Replacement algorithm. 10. Write a program to demonstrate the use of Optimal Page Replacement algorithm. 11. Write a program to demonstrate the use of FCFS Disk Scheduling algorithm. 12. Write a program to demonstrate the use of the Deadlock detection algorithm. 13. Write a program to demonstrate the use of SCAN Disk Scheduling algorithm. 14. Write a program to demonstrate the use of CSCAN Disk Scheduling algorithm. 15. Write a program to demonstrate the system calls of the operating system. 	
--	--	--

Elective-II

Course Code	2MCS6(1)	
Course Name	(1)Theory of Computation	
Total Credits	4	
Course Outcomes	<p>Course outcomes: Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. use basic concepts of formal languages of finite automata techniques 2. Design Finite Automata's for different Regular Expressions and Languages 3. Construct context free grammar for various languages 4. solve various problems of applying normal form techniques, push down automata and Turing Machines 	
Units	Contents	Total Hours
Unit I	Strings, alphabets and languages, Graphs and trees, Inductive proofs, set notations, relations, Finite automata and regular Expression: Finite state system, Non deterministic finite automata, Finite automata with ϵ -moves. Deterministic finite automata, equivalence between NFA and DFA, Conversion of NFA to DFA.	10

Unit II	Regular set and regular expression , Two way finite automata, finite automata with output, Applications of finite automata. Equivalence of RE and FA, Inter conversion, pumping lemma, closure property of regular sets, Regular grammars, Right linear and Left linear grammar, equivalence between Regular linear grammar and FA inter conversion between RE and RG.	10
Unit III	Context free grammar , derivation trees, Chomsky Normal Form, Greibach Normal Form. Push Down Automata: Definition, model, acceptance of CFL, equivalence of CFL and PDA, Interconversion, Enumeration of properties of CFL.	10
Unit IV	Turing Machine : Definition, model, Design of Turing machine, computable languages and function, Techniques of Turing machine construction, Modifications of Turing machine, Church's Hypothesis.	10
Unit V	Chomsky Hierarchy of languages , Linear bounded automata and context sensitive languages, Introduction of DCFL and DPDA, Decidability of problems.	10
Unit VI	Undecidability : Properties of recursive & non recursive enumerable languages, universal Turing machine, post correspondence problem, introduction to recursive function theory.	10
	Text books: 1. "Introduction to Automata theory, Languages and Computation"- Hopcraft J.E.& Ullman J.D.	
	Reference Books: 1. "An Introduction to Formal Languages and automata"- Peter Liz. 2. "Introductory theory of Computer Science"- V. Krishnamurthy(EWP) 3. "Elements of Theory & Computations"- Lavis and Padadimitron-PHI.	

Course Code	2MCS6(2)
Course Name	2 Computer System Architecture
Total Credits	04
Course Outcomes	<p>Course Outcome: Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. Understand the theory and architecture of central processing unit. 2. Analyze some of the design issues in terms of speed, technology, cost, performance. 3. Design a simple CPU with applying the theory concepts. 4. Use appropriate tools to design verify and test the CPU architecture. 5. apply the concepts of parallel processing, pipelining and interprocessor communication. 6. Understand the architecture and functionality of central processing unit. 7. Exemplify in a better way the I/O and memory organization.

	8. Define different number systems, binary addition and subtraction, 2's complement representation and operations with this representation.	
Units	Contents	Total Lectures
I	Instruction Set Architectures :Levels of Programming Languages, Language Categories ,Compiling and Assembling Programs, Assembly Language Instructions ,Instruction Types, Data Types ,Addressing Modes ,Instruction Formats, Instruction Set Architecture Design, A Relatively Simple Instruction Set Architecture , Introduction To Computer Organization : Basic Computer Organization , System Buses, Instruction Cycles, CPU Organization, Memory Subsystem Organization and Interfacing, Types of Memory, Internal Chip Organization , Memory Subsystem Configuration, I/O Subsystem Organization and Interfacing	10
II	Register Transfer Languages : Micro-Operations and Register Transfer Language, Using RTL to Specify Digital Systems, Specification of Digital Components, Specification and implementation of Simple System, More Complex Digital Systems and RTL, Module 6 Counter, Toll Booth Controller, CPU DESIGN : Specifying a CPU, Design and Implementation of a Very Simple CPU, Specifications for a Very Simple CPU ,Fetching Instructions from Memory, Design and Implementation of a Relatively Simple CPU, Specifications for a Relatively Simple CPU ,Fetching and Decoding Instructions, Executing Instructions, Establishing Data Paths, Design of a Relatively Simple ALU, Designing the Control Unit Using Hardwired Control, Design Verification	10
III	Microsequencer Control Unit Design : Basic Microsequencer Design, Microsequencer Operations, Microinstruction Formats, Design and Implementation of a Very Simple Microsequencer, The Basic Layout, Generating the Correct Sequence and Designing the Mapping Logic , Generating the Micro-Operations Using Horizontal Microcode, Generating the Micro-Operations Using Vertical Microcode, Design and Implementation of a Relatively Simple Microsequencer, Modifying the State Diagram, Designing the Sequencing Hardware and Microcode, Completing the Design Using Horizontal Microcode, Reducing the Number of Microinstruction, Microsubroutines, Microcode Jumps, Microprogrammed Control vs. Hardwired Control, Complexity of the instruction Set, Ease of Modification ,Clock Speed	10
IV	Computer Arithmetic : Unsigned Notation, Addition and Subtraction, Multiplication, Division, Signed Notation, Signed-Magnitude Notation, Signed-Two's Complement Notation, Binary Coded Decimal, BCD Numeric Format , Addition and Subtraction, Multiplication and Division, Floating Point Numbers, Numeric Format ,Numeric Characteristics, Addition and Subtraction, Multiplication and Division, Memory Organization : Hierarchical Memory Systems, Cache Memory ,Associative Memory , Cache Memory with Associative Mapping ,Cache Memory with Direct Mapping , Cache Memory with Set-Associative	10

	Mapping, Virtual Memory, Paging, Segmentation, Memory Protection	
V	<p>Input/Output Organization: Asynchronous Data Transfers, Source- initiated Data Transfer, Destination-Initiated Data Transfer, Handshaking, Programmed I/O, New Instruction, New Control Signals, New States and RTL Code, Modify the CPU Hardware for the New Instruction, Make Sure Other Instructions Still Work, Interrupts, Transferring Data Between the CPU and I/O Devices , Types of interrupts, Processing interrupts, Interrupt Hardware and Priority, Implementing interrupts inside the CPU, Direct Memory Access , Incorporating Direct Memory Access into a Computer System ,DMA Transfer Modes, Modifying the CPU to Work with DAM, I/O Processors ,Serial Communication, Serial Communication Basics ,Universal Asynchronous Receiver/ Transmitters (UARTs)</p>	10
VI	<p>Reduced Instruction Set Computing : RISC Rationale ,Fixed Length Instructions, Limited Loading and Storing Instruction Access Memory, Fewer Addressing Modes, Instruction Pipeline , RISC Instruction Sets, Instruction Pipelines and Register Windows, Instruction Pipelines ,Register Windowing and Renaming, Instruction Pipeline Conflicts ,Data Conflicts, Branch Conflicts, Introduction To Parallel Processing :Parallelism in Uniprocessor Systems, Organization of Multiprocessor Systems , Flynn's Classification System Topologies , MIMD System Architectures Communication in Multiprocessor Systems, Fixed Connections, Reconfigurable Connections, Routing on Multistage Interconnection Networks</p>	10
	<p>Text Books: Computer System Organization & Architecture – Johnd. Carpinelli,Publisher Pearson- Addison Wesley Longman</p>	
	<p>Reference Books: 1. Computer Fundamentals Architecture and Organization” by Ram B 2. Fundamentals of Computer Organization and Architecture (Wiley Series on Parallel and Distributed Computing)” by Mostafa Abd–El–Barr and Hesham El–Rewini 3. Fundamental of Computer Organization and Design” by Sivarama P Dandamudi 4. Computer Fundamentals, Third Edition: Architecture and Organization” by Ram B</p>	

Course Code	2MCS6(3)	
Course Name	(3)Enterprise Resource Management	
Credits	04	
Course Outcomes (COs)	<p>Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> (1) build an understanding of the fundamental concepts of ERP systems, their architecture, and working of different modules in ERP. (2) develop and design the modules used in ERP systems, and can customize the existing modules of ERP systems. (3) Analyze the strategic options for ERP identification and adoption. (4) Design the ERP implementation strategies. (5) Apply reengineered business processes for successful ERP implementation. (6) Implement ERP system with different approaches 	
Units	Contents	Total Hrs
I	<p>Enterprise Resource Planning- An Introduction: Defining Enterprise Resource Planning, Functional Modules in ERP Systems, Evolution of ERP Systems, Characteristics of ERP, Process Integration with ERP Systems, Benefits of ERP Applications, Technology Behind ERP Systems, Implementation Costs, Challenges Faced During ERP Implementation, Facts about ERP Implementation, Examples of ERP Implementation in India.</p>	10
II	<p>ERP Market and Vendors: ERP Market, ERP Vendors, Services-Oriented Architecture, ERP Package Features: Comparison and Selection Criteria, ERP Packages: The Big Players.</p> <p>Extended ERP Services: Defining Extended ERP, SCM and ERP, ERP and BI, ERP and E-commerce.</p>	10
III	<p>Business Process Re-engineering and ERP: Defining Business Process Re-engineering, Enterprise Redesign Principles, Business Process Re-engineering (Process Innovation) versus Total Quality Management (Process Improvement), BPR and Change Management, Different Approaches in BPR Implementation, Methodology for BPR Implementation, Role of IT in BPR, BPR and ERP systems, BPR Success/ Failure Factors, BPR Implementation Cases.</p>	10
IV	<p>Planning for ERP: Planning for ERP Implementation, Understanding organizational Requirements, Understanding Economic and Strategic Justification, Analysing Project Scope and Broad Implementation Approach, Determining Resources, Comprehending Top Management Commitment, Realizing Organizational Commitment to Change and Implementation,</p>	10
V	<p>Planning : Matching Business Processes with the Right ERP Systems, Creating a Budget for ERP Implementation, Selecting the Right ERP Packages, Preparing Organizations for ERP Implementation.</p> <p>Implementation of ERP:</p>	10

	Design of ERP Systems, ERP Implementation Approaches, ERP Implementation Life Cycle, Examples: ERP Implementation Life Cycle	
VI	<p>Managing ERP Projects: Risk/ Failure Factors in ERP Implementation, Examples of ERP failure, Mitigating Implementation Risks: Critical Success Factors, Management and Complexity of Large-scale ERP Projects, Training Users to Use ERP Systems, Evaluating ERP Projects.</p>	10
	<p>Text Book : Enterprise Resource Planning by Ashim Raj Singla,CENGAGE Learning</p> <p>Reference Book :</p> <ol style="list-style-type: none"> 1) Concept of Enterprise Resource Planning, Third Edition, by Ellen F.Monk and Bret J. Wagner, CENGAGE Learning. 2) Enterprise Resource Planning by Alexis Leon, Tata McGraw-Hill Education 3) Enterprise Resource Planning, Fourth Edition by Alexis Leon, Tata McGraw-Hill 4) Enterprise Resource Planning: Text & Cases by Rajesh Ray, Tata McGraw-Hill 	

Course Code	2MCS6(4)	
Course Name	(4)Mobile Computing	
Total Credits	4	
Course Outcomes	<p>Course Outcome: Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. Explain the basics of mobile Computing 2. Describe the functionality of Mobile IP and Transport Layer 3. Classify different types of mobile telecommunication systems 4. Demonstrate the Adhoc networks concepts and its routing protocols 5. Make use of mobile operating systems in developing mobile applications 	
Units	Contents	Hrs
I	<p>Mobile Communications: An Overview 1 Mobile Communications, Mobile Computing, Mobile Computing Architecture, Mobile Devices, Mobile System Networks, Data Dissemination, Mobility Management, Security.</p> <p>Mobile Devices and Systems Mobile Phones, Digital Music Players, Handheld Pocket Computers, Handheld Devices: Operating System, Smart Systems, Limitations of Mobile Devices, Automotive Systems</p>	10
II	<p>GSM-Services and Similar Architecture GSM-Services and System Architecture, Radio Interfaces, Protocols, Radio Interfaces , Localization, Calling Handover, Security, New Data Services</p>	10

	,General Packet Radio Service , High-speed Circuit Switched Data, DECT Wireless Medium Access Control and CDMA-based Communication Medium Access Control ,Introduction to CDMA-based Systems, Spread Spectrum in CDMA Systems ,Coding Methods in CDMA IST, IS-105 cdma One System, IMT-2000, i-mode, OFDM	
III	<p>Mobile IP Network Layer Mobile IP Network Layer, IP and Mobile IP Network Layers ,Packet Delivery and Handover Management , Location Management, Registration, Tunneling and Encapsulation, Route Optimization, Dynamic Host Configuration Protocol</p> <p>Mobile Transport Layer Conventional TCP/IP Transport Layer Protocols , Indirect TCP , Snooping TCP Mobile TCP, Other Methods of TCP-layer Transmission for Mobile, Networks TCP Over 2.5G/3G Mobile Networks</p>	10
IV	<p>Databases Database Hoarding Techniques, Data Caching, Client-Server Computing and Adaptation ,Transactional Models, Query Processing, Data Recovery Process Issues relating to Quality of Service</p> <p>Data Dissemination and Broadcasting Systems Communication Asymmetry, Classification of Data-Delivery Mechanisms, Data Dissemination Broadcast Models ,Selective Tuning and Indexing Techniques, Digital Audio Broadcasting,</p>	10
V	<p>Data Synchronization in Mobile Computing System Synchronization, Synchronization Software for Mobile Devices, Synchronization Protocols, SyncML-Synchronization Language for Mobile Computing Sync4J (Funambol) ,Synchronized Multimedia Markup Language (SMIL)</p> <p>Mobile Devices: Server and Management Mobile Agent , Application Server, Gateways ,Portals ,Service Discovery , Device Management, Mobile File Systems, Security</p>	10
VI	<p>Mobile Ad-hoc and Sensor Networks Introduction to Mobile Ad-hoc Network , MANET, Wireless Sensor Networks, Applications Mobile Application Languages –XML Java,J2ME,and Java Card Introduction ,XML, JAVA, Java 2 Micro Edition (J2ME) ,JavaCard Mobile OperatingSystems Operating System, PalmOS, WindowsCE, Symbian OS, Linux for Mobile Devices</p>	10
	<p>Text Books: Mobile Computing: Raj Kamal (Oxford)</p>	
	<p>Reference Books: 1) Mobile Communication: Jochen Schiller (PE) 2) Principles of mobile communication: Gordon L. Stuber (Springer) 3) Wireless Communications: Principles and Practice: Theodore S. Rappaport(Pearson)</p>	

Course Code	2MCS6(5)	
Course Name	(5)Compiler Construction	
Total Credits	4	
Course Outcomes	<p>Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. explore the principles, algorithms, and data structures involved in the design and construction of compilers. 2. write the machine dependent code 3. draw the flow graph for the intermediate codes. 4. Design a compiler for a simple programming language. 	
Units	Contents	Total Hours
Unit I	Introduction to Compilers: Overview, typical compiler Structure, implementation. Programming Language Grammars: Elements of formal language grammars, derivation, reduction, syntax tree, ambiguity, regular grammars and expressions.	10
Unit II	Scanning and Parsing Techniques: The scanner, top-down and bottom-up parsing, syntax directed translation, Symbol table organization, Hash table organization, Linked List and Tree structured symbol tables, symbol table organization for structures and records.	10
Unit III	Memory Allocation: Static and dynamic memory allocation, array allocation and access, allocation for strings, structure allocation, common and equivalence allocation. Compilation of expressions.	10
Unit IV	Compilation of control structures: Control transfers, procedural calls, conditional execution, iteration control constructs.	10
Unit V	Error detection, indication and recovery. Compilation of I/O statements: Compilation of I/O list, compilation of FORMAT list, the I/O routine, file control.	10
Unit VI	Code optimization: Major issues, optimizing transformations, local optimizations, program flow analysis, Global optimization, writing compilers.	10
	Text books: 1. Compiler construction – D.M. Dhamdhere, Macmillan India Ltd.	
	Reference Books: 1. Principles of Compiler Design – Alfred V. Aho, Jeffrey D. Ullman 2. The Theory and Practice of Compiler Writing – J.P. Trembly, P.G. Sorenson McGraw Hill Publication. 3. Engineering a compiler – K.D. Cooper and Linda Torczon, Elsevier Direct Publ.	

Laboratories

Course Code	2MCS7
Course Name	Lab-III 3,4-SE-Tools/DM Tools
Credits	2
Course Outcomes	<p style="text-align: center;">Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. Identify different actors and use cases from a given problem statement and draw use case diagram to associate use cases with different types of relationship. 2. Draw a class diagram after identifying classes and association among them. 3. Graphically represent various UML diagrams and associations among them and identify the logical sequence of activities undergoing in a system, and represent them pictorially. 4. use modern engineering tools for specification, design, implementation and testing 5. translate end-user requirements into system and software requirements 6. generate a high-level design of the system from the software requirements. 7. Provide a formal basis for understanding the modeling language. 8. apply standard data mining methods and techniques such as association rules, data clustering and classification. 9. Learn new, advanced techniques for emerging applications (e.g. social network analysis, stream data mining). 10. Gain practical intuition about how to apply these techniques on datasets of realistic sizes using modern data analysis frameworks.

Sr. No.	Contents	Total Hrs
<p>The following list of can be used as guidelines for basic understanding but the scope of the laboratory should not be limited to this list. Aim of the list is to inform about minimum expected outcomes.</p>		
1	To prepare PROBLEM STATEMENT for any project.	
2	Identifying the Requirements from Problem Statements.	
3	Estimation of Project Metrics.	
4	Modeling UML Use Case Diagrams and Capturing Use Case Scenarios.	
5	Estimation of Test Coverage Metrics and Structural Complexity.	
6	Designing Test Suites.	

7	To draw a sample ENTITY RELATIONSHIP DIAGRAM for real project or system by using IBM Rational Rose software.	
8	To draw the Use Case Diagram using Rational Rose.	
10	To draw a sample activity diagram for real project or System IBM RationalRose software.	
10	To prepare STATE CHART DIAGRAM for any project IBM Rational Rose Software.	
11	To draw the Sequence Diagram using IBM Rational Rose software.	
12	To draw the collaboration Diagram using IBM Rational Rose software.	
13	To draw class diagram for any project by using IBM Rational Rose Software	
14	Create a use- case diagram for tour management system and library domain model by using Dia software.	
15	To create an ER diagram for hotel management system by using Dia software.	
16	To create a component diagram for Bank management system by using Diasoftware.	
17	Create a use- case diagram for tour management by using Diasoftware.	
18	To create a library domain model by using Dia software.	
110	To find out complexities & create route testing for ATM transaction using Dia software.	
20	To test load with the help of burst strategy in Soap UI tool.	
21	To perform the load testing with thread strategy in Soap UI tool.	
22	To perform a simple load testing by using Soap UI tool.	
23	To measures the load with variance strategy in Soap UI tool.	
24	Study and usage of any Design phase CASE tool	
25	Performing the Design by using any Design phase CASE tools.	
26	Develop test cases for unit testing and integration testing.	
27	Prepare a SRS document in line with the IEEE recommended standards.	
	<p>Data mining Tools used : Weka / Rapid miner</p> <ol style="list-style-type: none"> 1. Create an Employee Table with the help of Data Mining Tool. Description: We need to create an Employee Table with training data set which includes attributes like name, id, salary, experience, gender, phone number. 2. Create a Weather Table with the help of Data Mining Tool. Description: We need to create a Weather table with training data set which includes attributes like outlook, temperature, humidity, windy, play. 3. Apply Pre-Processing techniques to the training data set of Weather Table 	

	<p>Description: Real world databases are highly influenced to noise, missing and inconsistency due to their queue size so the data can be pre-processed to improve the quality of data and missing results and it also improves the efficiency. There are 3 pre- processing techniques they are:</p> <ol style="list-style-type: none"> 1) Add 2) Remove 3) Normalization <p>4. Apply Pre-Processing techniques to the training data set of Employee Table</p> <p>Description: Real world databases are highly influenced to noise, missing and inconsistency due to their queue size so the data can be pre-processed to improve the quality of data and missing results and it also improves the efficiency. There are 3 pre- processing techniques they are:</p> <ol style="list-style-type: none"> 1) Add 2) Remove 3) Normalization <p>5. Normalize Weather Table data using Knowledge Flow.</p> <p>Description: The knowledge flow provides an alternative way to the explorer as a graphical front end to WEKA's algorithm. Knowledge flow is a working progress. So, some of the functionality from explorer is not yet available. So, on the other hand there are the things that can be done in knowledge flow, but not in explorer. Knowledge flow presents a dataflow interface to WEKA. The user can select WEKA components from a toolbar placed them on a layout campus and connect them together in order to form a knowledge flow for processing and analyzing the data.</p> <p>6. Normalize Employee Table data using Knowledge Flow.</p> <p>Description: The knowledge flow provides an alternative way to the explorer as a graphical front end to WEKA's algorithm. Knowledge flow is a working progress. So, some of the functionality from explorer is not yet available. So, on the other hand there are the things that can be done in knowledge flow, but not in explorer. Knowledge flow presents a dataflow interface to WEKA. The user can select WEKA components from a toolbar placed them on a layout campus and connect them together in order to form a knowledge flow for</p>	
--	--	--

	<p>processing and analyzing the data.</p> <p>7. Finding Association Rules for Buying data. Description: In data mining, association rule learning is a popular and well researched method for discovering interesting relations between variables in large databases. It can be described as analyzing and presenting strong rules discovered in databases using different measures of interestingness. In market basket analysis association rules are used and they are also employed in many application areas including Web usage mining, intrusion detection and bioinformatics.</p> <p>8. Finding Association Rules for Banking data. Description: In data mining, association rule learning is a popular and well researched method for discovering interesting relations between variables in large databases. It can be described as analyzing and presenting strong rules discovered in databases using different measures of interestingness. In market basket analysis association rules are used and they are also employed in many application areas including Web usage mining, intrusion detection and bioinformatics.</p> <p>9. Finding Association Rules for Employee data. Description: In data mining, association rule learning is a popular and well researched method for discovering interesting relations between variables in large databases. It can be described as analyzing and presenting strong rules discovered in databases using different measures of interestingness. In market basket analysis association rules are used and they are also employed in many application areas including Web usage mining, intrusion detection and bioinformatics.</p> <p>10. To Construct Decision Tree for Weather data and classify it. Description: Classification & Prediction: Classification is the process for finding a model that describes the data values and concepts for the purpose of Prediction. Decision Tree: A decision Tree is a classification scheme to generate a tree consisting of root node, internal nodes and external nodes. Root nodes representing the attributes. Internal nodes are also the attributes. External nodes are the classes and each branch represents the values of the attributes Decision Tree also contains set of rules for a given data set;</p>	
--	--	--

	<p>there are two subsets in Decision Tree. One is a Training dataset and second one is a Testing data set. Training data set is previously classified data. Testing data set is newly generated data.</p> <p>11. To Construct Decision Tree for Customer data and classify it. Description: Classification & Prediction: Classification is the process for finding a model that describes the data values and concepts for the purpose of Prediction. Decision Tree: A decision Tree is a classification scheme to generate a tree consisting of root node, internal nodes and external nodes. Root nodes representing the attributes. Internal nodes are also the attributes. External nodes are the classes and each branch represents the values of the attributes Decision Tree also contains set of rules for a given data set; there are two subsets in Decision Tree. One is a Training dataset and second one is a Testing data set. Training data set is previously classified data. Testing data set is newly generated data.</p> <p>12. To Construct Decision Tree for Location data and classify it. Description: Classification & Prediction: Classification is the process for finding a model that describes the data values and concepts for the purpose of Prediction. Decision Tree: A decision Tree is a classification scheme to generate a tree consisting of root node, internal nodes and external nodes. Root nodes representing the attributes. Internal nodes are also the attributes. External nodes are the classes and each branch represents the values of the attributes Decision Tree also contains set of rules for a given data set; there are two subsets in Decision Tree. One is a Training dataset and second one is a Testing data set. Training data set is previously classified data. Testing data set is newly generated data.</p> <p>13. Write a procedure for Visualization for Weather Table. Description: This program calculates and has comparisons on the data set selection of attributes and methods of manipulations have been chosen. The Visualization can be shown in a 2-D representation of the information.</p> <p>14. Write a procedure for Visualization of Banking Table.</p>	
--	---	--

	<p>Description: This program calculates and has comparisons on the data set selection of attributes and methods of manipulations have been chosen. The Visualization can be shown in a 2-D representation of the information.</p> <p>15. Write a procedure for cross-validation using J48 Algorithm for weather table.</p> <p>Description: Cross-validation, sometimes called rotation estimation, is a technique for assessing how the results of a statistical analysis will generalize to an independent data set. It is mainly used in settings where the goal is prediction, and one wants to estimate how accurately a predictive model will perform in practice. One round of cross-validation involves partitioning a sample of data into complementary subsets, performing the analysis on one subset (called the <i>training set</i>), and validating the analysis on the other subset (called the <i>validation set</i> or <i>testing set</i>).</p>	
--	--	--

Course Code	2MCS8
Course Name	Lab-IV 2 Graphic Programming & Tools
Credits	2
Course Outcomes	<p>Upon completion of this course successfully, students would be able to</p> <ol style="list-style-type: none"> 1. Apply the basic concepts of C programming in developing the code for various Graphics applications. 2. Individually Apply C programming concepts to do the Programming for Computer Graphics. 3. Identify and apply a suitable transformation for a given application.
	<p>* List of Practical</p> <ol style="list-style-type: none"> 1. Write a program to draw a line using Bresenham's line drawing algorithm. 2. Write a program to draw a line using DDA line drawing algorithm. 3. Write a program to draw a circle using ARCDNA algorithm. 4. Write a program to draw a rectangle. 5. Write a program to draw an ellipse using ARCDNA algorithm. 6. Write a program to rotate an object about origin. 7. Write a program to rotate an object about any arbitrary point. 8. Write a program to implement 2D Scaling Transformation. 9. Write a program to implement 3D Scaling Transformation. 10. Write a program to implement 2D Translation Transformation.

	<ol style="list-style-type: none"> 11. Write a program to implement 3D Translation Transformation. 12. Write a program to move an image (ball) on the screen. 13. Write a program to generate a Cubic Bezier Curve. 14. Write a program to draw a Polygon using Absolute and Relative commands. 15. Write a program to clip user defined area of a screen. 16. Write a program to clip line segment. 17. Write a program to clip polygon. 18. Write a program to demonstrate rotation of point. 19. Write a program to fill area by given pattern. 20. Write a program to fill the closed area using flood fill method. 21. Write a program to fill the closed area using Scan line method. 22. Write a program to draw a line using simple DDA line drawing algorithm.
--	---

GIC

Course Name		GIC3: Webpage Design Techniques
Units	Contents	
Unit I	The Environment and the Tools: <ul style="list-style-type: none"> • Introduction • The Internet and the World Wide Web • Ways to Access the Internet • Types of Websites • Web Design Tools • Web Design Roles 	
Unit II	Web Publishing Fundamentals: <ul style="list-style-type: none"> • Introduction • Advantages of Web Publishing • Basic Web Design Principles • Color as Web Design Tool • Writing for the Web • Web Publishing Issues 	
Unit III	Planning a Successful Website: <ul style="list-style-type: none"> • Introduction • The Website Development Planning Process • Step 1: Define the Website's Purpose and Audience • Step 2: Determine the Websites General Content • Step 3: Select the Websites Structure 	
Reference Book: Introductory Web Design by Jennifer T. Campbell, CENGAGE Publication		

Course Name		GIC4: Automation with Robotics
Units	Contents	

Unit I	<p>FUNDAMENTALS OF ROBOT: Robotics-Philosophical Considerations, Definition and Advantages of Robotics, Laws of Robotics, Motivating Factors for the Introduction of Robotics System to the Industrial World, Robot, History of Robots, Objectives of using Industrial Robots, Advantages and Disadvantages of Robots, Types of Industrial Robots., Robotic Systems, Robot Classifications, Mechanical Design of a Robot, Types of Mechanical Joints, Robot Arms, Robot Hands, Robots Qualities, Robot Specifications, Robots Performance Testing</p>
Unit II	<p>ROBOTICS OPERATIONS AND CONTROL:</p> <ul style="list-style-type: none"> • Robots Kinematic Control: Robot arm kinematics, Robot arm dynamics, Kinematic chains - The manipulator, Robot control systems, Types of controllers used in robotics, Trajectory Planning and motion control manipulator • Conversion of Motion • Techniques of Robot Calibration • Robot Sensing and Vision • Robot Programming Languages
Unit III	<p>INDUSTRIAL AUTOMATION: General Aspects, Advantages and Limitations of Automation, Applications of Automation, Elements of Automation, Aims of Automation, Mechanization and Automation, Types of Automation, Low Cost Automation, Assembly Automation Equipment - Transfer Devices and Feeders, Flexible Manufacturing System (FMS), Computer Integrated Manufacturing (CIM), Mechatronics and Concurrent Engineering, Computer Aided Process Planning (CAPP) system, Group Technology</p>
<p>Reference Book: Robotics and Industrial Automation by R. K. Rajput S. CHAND Publication</p>	

Appendix-A

Master of Science (M.Sc. Computer Science) Full Time Two Years Degree Programme :-

- 1) A Student shall have to be admitted every year in the respective Institute/ College/University Department for completion of an academic year of this two year Degree program.
- 2) The M.Sc. Degree shall consist of four semesters i.e. Semester I & II in the first academic year, Semester III & IV in the second academic year.
- 3) Student has to complete all four Semesters successfully for the award of Degree of Master of Science and fulfill conditions as per Ordinance No. 19.
- 4) Every Semester of M.Sc. Computer Science Program shall be of at least 90 teaching days in a semester and shall be of at least 180 teaching days in an academic year.
- 5) The Examinations shall consist of the subjects as indicated in the Scheme of Examinations as per **Appendix – ‘A1 to A4’**.
- 6) The Semester wise structure of the program is as follows.

Sant Gadge Baba Amravati University Amravati
Scheme of Teaching, Learning & Examination leading to the Degree Master of Science (Computer Science)
(Two Years- Four Semesters Degree Programme- C.B.C.S)
(M. Sc. Part-I) Semester- I

S. N.	Subject Code	Type	Subject	Teaching & Learning Scheme						Duration Of Exam Hours	Examination & Evaluation Scheme								
				Teaching Period Per Week							Credits			Maximum Marks				Minimum Passing	
				L	T	P	Total	L / T	Practical		Total	Theory		Practical		Total Marks	Marks	Grade	
												Theory + MCQ External	Theory Internal	Internal	External				
Core Subject																			
1	1MCS1	DSC1	1 Computer System Design	4	-	-	4	4	-	4	3	80	20			100	40	P	
2	1MCS2	DSC2	2 Data Structure with OOP	4	-	-	4	4	-	4	3	80	20			100	40	P	
3	1MCS3	DSC3	3 Data Base Management Technologies	4	-	-	4	4	-	4	3	80	20			100	40	P	
4	1MCS4	DSC4	4 Computer Network & Wireless Technology	4	-	-	4	4	-	4	3	80	20			100	40	P	
Skill-1																			
5	1MCS5	SEC1	4-Advanced Java/ NS2/ tools	-	2	2	4	4	-	4	3	-	-	25	25	50	25	P	
Elective-1																			
6.	1MCS6(1)	DSE1	(1) Discrete Mathematical Structure																
7.	1MCS6(2)	DSE2	(2) Entrepreneurship Development																
8.	1MCS6(3)	DSE3	(3) Research Methodology	4	-	-	4	4	-	4	3	80	20			100	40	P	
9.	1MCS6(4)	DSE4	(4) Management Information System																
10.	1MCS6(5)	DSE5	(5) Data Science and Analytics																
Laboratories																			
11	1MCS7	Lab-I	1,2 - Programming(C/C++/Java/ALP)	-	-	4	4	-	2	2	3	-	-	25	25	50	25	P	
12	1MCS8	Lab-II	3-SQL/ DBMS tools, MSsql, My Sql	-	-	4	4	-	2	2	3	-	-	25	25	50	25	P	
Internship																			
13	1MCS9		#Internship/Field Work/Work Experience@																
			TOTAL	20	2	10	32												
Open Elective(Apendix-5)																			
14	1MCS10	OE1	Open elective (OE) /GIC/Open skill/MOOC*	-	2	-	2	-	1	1	-			25	25	50	25	P	
			TOTAL	20	4	10	34	24	5	29									
GIC																			
		GIC1	User Experience Deign																
		GIC2	Effective Email Communication																

L: Lecture, T: Tutorial, P: Practical

Students may complete their Internship/Field Work/Work Experience in First OR Second OR Third Semester of M. Sc. (Computer Science) according to their convenience; @ denotes Ancillary Credit

Note : Internship /Apprenticeship/Field Work / Work Experience (During vacations of Semester I to Semester III) for duration of minimum 60 hours to maximum 90 hours mandatory to all the students, to be completed during vacations of Semester I to III. This will carry 2 Credits for learning of 60 hours or 3 Credits for learning of 90 hours. Its credits and grades will be reflected in final semester IV credit grade report.

- OEC can be studied during semester I to IV-As per Appendix 5

Sant Gadge Baba Amravati University Amravati
Scheme of Teaching, Learning & Examination leading to the Degree Master of Science (Computer Science)
(Two Years- Four Semesters Degree Programme- C.B.C.S)
(M. Sc. Part-I) Semester- II

S. N.	Subject Code	Type	Subject	Teaching & Learning Scheme						Duration of Exam Hours	Examination & Evaluation Scheme									
				Teaching Period Per Week							Credits			Maximum Marks					Minimum Passing	
				L	T	P	Total	L / T	Practical		Total	Theory		Practical		Total Marks	Marks	Grade		
												Theory + MCQ External	Theory Internal	Internal	External					
Core Subject																				
1	2MCS1	DSC1	1 Operating System Algorithms	4	-	-	4	4	-	4	3	80	20			100	40	P		
2	2MCS2	DSC2	2 Graphics Application programming	4	-	-	4	4	-	4	3	80	20			100	40	P		
3	2MCS3	DSC3	3 Software Engineering	4	-	-	4	4	-	4	3	80	20			100	40	P		
4	2MCS4	DSC4	4 Data Mining and Data Warehousing	4	-	-	4	4	-	4	3	80	20			100	40	P		
Skill-2																				
5	2MCS5	SEC2	1 -OS (Windows / Android /Linux)	-	2	2	4	4	-	4	3	-	-	25	25	50	25	P		
Elective-2																				
6.	2MCS6(1)	DSE1	(1)Theory of Computation	4	-	-	4	4	-	4	3	80	20			100	40	P		
7.	2MCS6(2)	DSE2	(2)Computer System Architecture																	
8,	2MCS6(3)	DSE3	(3)Enterprise Resource Management																	
9.	2MCS6(4)	DSE4	(4)Mobile Computing																	
10.	2MCS6(5)	DSE5	(5)Compiler Construction																	
Laboratories																				
11	2MCS7	Lab-III	3,4 -SE tools/ DM tools	-	-	4	4	-	2	2	3	-	-	25	25	50	25	P		
12	2MCS8	Lab-IV	2- Graphics programming and tools	-	-	4	4	-	2	2	3	-	-	25	25	50	25	P		
Internship																				
13	2MCS9		#Internship/Field Work/Work Experience@																	
			TOTAL	20	2	10	32													
Open Elective(Appendix 5)																				
14	2MCS10	OE2	Open elective (OE) /GIC/Openskill/MOOC*	-	2	-	2	-	1	1	-			25	25	50	25	P		
			TOTAL	20	4	10	34	24	5	29										
GIC																				
		GIC3	Web Page Design Techniques																	
		GIC4	Automation With Robotics																	

L: Lecture, T: Tutorial, P: Practical

Students may complete their Internship/Field Work/Work Experience in First OR Second OR Third Semester of M. Sc. (Computer Science) according to their convenience; @ denotes Ancillary Credit

Note : Internship /Apprenticeship/Field Work / Work Experience (During vacations of Semester I to Semester III) for duration of minimum 60 hours to maximum 90 hours mandatory to all the students, to be completed during vacations of Semester I to III. This will carry 2 Credits for learning of 60 hours or 3 Credits for learning of 90 hours. Its credits and grades will be reflected in final semester IV credit grade report.

- OEC can be studied during semester I to IV- As per Appendix 5

L: Lecture, T: Tutorial, P: Practical

Students may complete their Internship/Field Work/Work Experience in First OR Second OR Third Semester of M. Sc. (Computer Science) according to their convenience; @ denotes Ancillary Credit

Note : Internship /Apprenticeship/Field Work / Work Experience (During vacations of Semester I to Semester III) for duration of minimum 60 hours to maximum 90 hours mandatory to all the students, to be completed during vacations of Semester I to III. This will carry 2 Credits for learning of 60 hours or 3 Credits for learning of 90 hours. Its credits and grades will be reflected in final semester IV credit grade report.

- OEC can be studied during semester I to IV- As per Appendix 5

L: Lecture, T: Tutorial, P: Practical

Students may complete their Internship/Field Work/Work Experience in First OR Second OR Third Semester of M. Sc. (Computer Science) according to their convenience; @ denotes Ancillary Credit

Note : Internship /Apprenticeship/Field Work / Work Experience (During vacations of Semester I to Semester III) for duration of minimum 60 hours to maximum 90 hours mandatory to all the students, to be completed during vacations of Semester I to III. This will carry 2 Credits for learning of 60 hours or 3 Credits for learning of 90 hours. Its credits and grades will be reflected in final semester IV credit grade report.

- OEC can be studied during semester I to IV-As per Appendix 5

Total Credits: 119

Appendix- 5

Common Instructions for all the Semesters regarding Choice Based Credits (CBC)/Open Electives (OE) are as under:

The titles of broad activity those can be undertaken by the students in every semester and their respective credits are listed in the table given below. Student has to undertake one or more activities out of these table so as to avail at least 2 credits per semester

The Subjects/Modules Activity to be undertaken by the Student under the Open Electives approved by the Department Institute. The schedule of approval will be declared by the Department/Institute at the beginning of the Semester (1 July) as per details given below:

One Faculty Member will work as a Coordinator for Open Electives for which 01 Hour of Theory period will be considered as a weekly work load against this work. All Coordinators has to do counselling of respective Open electives, do the Students Registration process and allot them to faculty members (will be working as a mentor). All these electives are internally accessed by respective Coordinators & Guides based on Minimum 03 Class Tests/ Final Objective Test/ Demo/ Report Submission/Certificate issued by competent authority Viva Voce and other methods as decided by the Department/Institute.

The Mentor shall conduct Tutorial Classes for Workload counting purpose, it should be noted that: 01 Tutorial hour is equal to 01 Theory Hour. For Tutorial, Batch of Maximum 20 Students will be considered and the Tutorial Batch should not be comprised of Less than 04 Students.

Coordinator shall take care that the students are not repetitively opting for same type of Electives in every Semester.

Summary of conduction of Choice Based Credits (CBC)/Open Electives (OC) Electives for all Semesters

- i. Electives Selection Process starts at beginning of the Semester,
- ii. Declare the names of Coordinator for Open Electives.
- iii. Counselling of Students by Coordinators for selection of Open Electives
- iv. Registration of Students by Coordinators under respective Open Electives
- v. Allotment of Registered Students to Mentor from Department.
- vi. Guidance/Counselling to Students by Mentor throughout the Semester
- vii. General Counselling by Coordinators over the Semester, whenever required.
- viii. Final Assessment of Students by Coordinators & Mentor for Allotment of Final Credits
- ix. Submission of Credits gained by Students to the Head of Department from Coordinators

The Open Elective and Credit Assigned	Credit
Successful completion of Online Course of 4 weeks	4
Project activity	4
Seminar Activity	1
Paper/poster presentation	1
Completion of soft skill programme of one week	1
Internship of 30 Hrs	2
Field Visit of 15 Hrs	1
Startup recognized and approved by the department	2

Participation in Unnat Bharat Abhiyan	1 for 15days, maximum 4
Yoga Meditation camp of 1 week	1
Completion of course/activity of similar credits proposed by the department from among the available courses/activities from other department/faculty in the college/university	4