CENTRAL UNIVERSITY OF HARYANA
MAHENDERGARH

Department of Computer Science & Information Technology

(Scheme and Syllabi of Master of Computer Applications (MCA)-2years Programme)
With effect from the Academic Session 2020-21 as per the latest AICTE guidelines

Total Credit = 104
Semester Wise Distribution of Credits: 27 + 27 + 27 + 23

Eligibility for Admission to MCA 2-year Programme:

Passed BCA/ Bachelor Degree in Computer Science and Engineering or equivalent Degree.

OR

Passed B.Sc./ B.Com./ B.A. with Mathematics at 10+2 Level or at Graduation Level. Obtained at least 50% marks (45% marks in case of candidates belonging to reserved category) in the qualifying Examination. ***The students admitted with this eligibility will have to simultaneously undertake additional Bridge Course(s) as prescribed by the University during the first semester.***

Bridge Course (Non Credit Course)

S.No	Course Name	Course Code	L	T	P	Credit
1	Fundamentals of Computer Science	SBS CS 01 01 01 E 3104	3	1	0	0
2	Internet Fundamentals	SBS CS 01 01 02 E 3104	3	1	0	0
3	Computer Programming using C	SBS CS 01 01 03 E 3104	3	1	0	0
4	Computer Programming using C -Lab-I	SBS CS 01 01 04 E 0021	0	0	2	0
					Total	0

Note: It is compulsory for each student to pass out Bridge Course(s) during the first semester only (three additional theory papers and one practical as prescribed in scheme of Bridge Course). Papers under Bridge Course will be taught only in the 1st semester of the MCA programme.

MCA 1st Semester

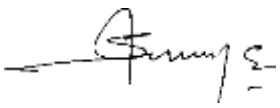
S.No	Course Name	Course Code	L	T	P	Credit
1	Data Structures	SBS CS 01 01 01 C 4004	4	0	0	4
2	Computer Networks	SBS CS 01 01 02 C 4004	4	0	0	4
3	Computer Organization and Architecture	SBS CS 01 01 03 C 4004	4	0	0	4
4	Discrete Mathematical Structures	SBS CS 01 01 04 C 4004	4	0	0	4
5	Operating System and Shell Programming	SBS CS 01 01 05 C 4004	4	0	0	4
6	**General Elective Course-I (To be taken from another department)**		4	0	0	4
7	Data Structures using C Lab-I	SBS CS 01 01 06 C 0021	0	0	2	1
8	Operating System and Shell Programming Lab -II	SBS CS 01 01 07 C 0021	0	0	2	1
9	Web Development using PHP Lab-III	SBS CS 01 01 08 C 0021	0	0	2	1
					Total	**27**

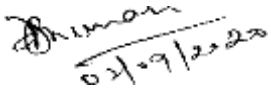
Students have to under go two weeks in house training on (any platform or software) during winter vacations and prepare its report which will be evaluated in 2nd semester.

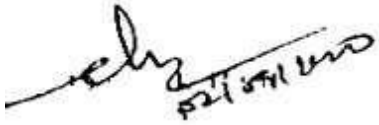
MCA 2nd Semester

S. No	Course Name	Course Code	L	T	P	Credit
1	Database Management System	SBS CS 01 02 09 C 4004	4	0	0	4
2	Computer Graphics	SBS CS 01 02 10 C 4004	4	0	0	4
3	Object Oriented Programming	SBS CS 01 02 11 C 4004	4	0	0	4
4	Software Engineering	SBS CS 01 02 12 C 4004	4	0	0	4
5	Design and Analysis of Algorithms	SBS CS 01 02 13 C 4004	4	0	0	4
6	**Departmental Elective Course-I**					
7	Database Management System Lab-I	SBS CS 01 02 14 C 0021	0	0	2	1
8	Object Oriented Programming using C++ Lab-II	SBS CS 01 02 15 C 0021	0	0	2	1
9	Design and Analysis of Algorithms Lab-III	SBS CS 01 02 16 C 0021	0	0	2	1
10	Evaluation of Open Source Training	SBS CS 01 02 17 C 0101	0	1	0	1
	List for Departmental Elective Courses-I					
	Mobile Communication	SBS CS 01 02 05 E 3003	3	0	0	3
	Management Information System and E-Commerce	SBS CS 01 02 06 E 3003	3	0	0	3
	Quantum Computing	SBS CS 01 02 07 E 3003	3	0	0	3
	Theory of Computation	SBS CS 01 02 08 E 3003	3	0	0	3
					Total	**27**

Students have to undergo four to six weeks training during summer vacations and prepare its report which will be evaluated in 3rd Semester.







MCA 3rd Semester

S. No	Course Name	Course Code	L	T	P	Credit
1	Complier Design	SBS CS 01 03 18 C 4004	4	0	0	4
2	Artificial Intelligence and Expert System	SBS CS 01 03 19 C 4004	4	0	0	4
3	Information and Network Security	SBS CS 01 03 20 C 4004	4	0	0	4
4	Internet and Java Programming	SBS CS 01 03 21 C 4004	4	0	0	4
5	**Departmental Elective Course-II**					
6	**General Elective Course-II (To be taken from another department)**		4	0	0	4
7	Internet and Java Programming Lab-I	SBS CS 01 03 22 C 0021	0	0	2	1
8	Artificial Intelligence with Python Lab-II	SBS CS 01 03 23 C 0021	0	0	2	1
9	Android Application Development Lab-III	SBS CS 01 03 24 C 0021	0	0	2	1
10	Evaluation of Training	SBS CS 01 03 25 C 0101	0	1	0	1
	List for Departmental Elective Courses-II					
	Network Programming	SBS CS 01 03 09 E 3003	3	0	0	3
	Machine and Deep Learning	SBS CS 01 03 10 E 3003	3	0	0	3
	Software Project Management	SBS CS 01 03 11 E 3003	3	0	0	3
	Digital Image Processing	SBS CS 01 03 12 E 3003	3	0	0	3
	Data Warehousing and Data Mining	SBS CS 01 03 13 E 3003	3	0	0	3
					Total	**27**

Students have to undergo open source skill enhancement training during winter vacations, and prepare its report which will be evaluated in 4th Semester.

MCA 4th Semester

S. No	Course Name	Course Code	L	T	P	Credit
1	Data Science with R Programming	SBS CS 01 04 26 C 4004	4	0	0	4
2	Wireless Sensor Network and Internet of Things	SBS CS 01 04 27 C 4004	4	0	0	4
3	**Departmental Elective Course-III**					
4	Data Science with R Programming Lab-I	SBS CS 01 04 28 C 0021	0	0	2	1
5	Advance Java Programming Lab-II	SBS CS 01 04 29 C 0021	0	0	2	1
6	Project Work (In House)	SBS CS 01 04 30 C 00168	0	0	16	8
7	Evaluation of Open Source Training	SBS CS 01 04 31 C 0101	0	1	0	1
	List for Departmental Elective Courses-III					
	Distributed and Cloud Computing	SBS CS 01 04 14 E 3104	3	1	0	4
	Bioinformatics	SBS CS 01 04 15 E 3104	3	1	0	4
	Natural Language Processing and Speech Recognition	SBS CS 01 04 16 E 3104	3	1	0	4
	Computer Vision	SBS CS 01 04 17 E 3104	3	1	0	4
	Embedded Programming	SBS CS 01 04 18 E 3104	3	1	0	4
					Total	**23**

Note: Practical work of all compulsory labs will be based on the corresponding theoretical course.

List of General Elective Papers (Offered to the other departments)

S. No	Course Name	Course code	L	T	P	Credit
1	Fundamentals of Computer Science	SBS CS 01 01 01 E 3104	3	1	0	4
2	Internet Fundamentals	SBS CS 01 01 02 E 3104	3	1	0	4
3	Emerging Trends and Technologies in IT	SBS CS 01 02 19 E 3104	3	1	0	4

Fundamentals of Computer Science
Course Code: SBS CS 01 01 01 E 3104

L T P **Total Credits: 0(Bridge Course)/ 4(GEC)**
3 1 0

Course Objectives:
This course aims to give students an in-depth understanding of why computers are essential components in business, education and society. This course will provide hands-on use of Microsoft Office applications Word, Excel, Access and PowerPoint.

Course Level Learning Outcomes:
Upon successful completion of the course students will be able to:

- Explain principal differences in various operating systems
- Identify computer systems technical specifications
- Describe the working of internet technologies
- Perform tasks like compose, format and edit a word document and other office software.

UNIT-I

Overview of Computer System: Evolution of Computer Systems, Generations of Computers, Parts of Computer System, Categories of Computers, Computer System Characteristics, Computer Hardware. Working of input & output devices: keyboard, mouse, trackball, pen, touch screens, scanner, digital camera, monitor, and printer. Working of storage devices: magnetic tape, magnetic disk, CD, DVD.Software- System & Application.

UNIT-II

Computer Memory: RAM ROM, EROM, Volatile and Non Volatile Memory, Types of Memory, Primary and Secondary Memory Input, Output & Storage devices, Organization of Secondary storage media.

UNIT-III

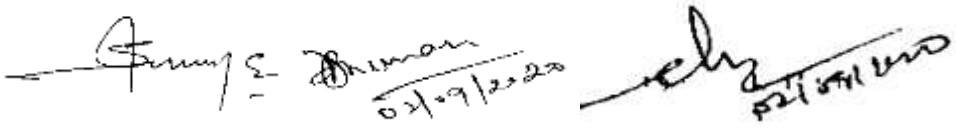
Operating System: Role of an OS, Types of OS, Features of OS & functions of OS. Booting Procedure.
Graphical OS: Fundamentals of windows, types of windows, Anatomy of Windows, Windows explorer, customizing windows, control panel, taskbar setting.

UNIT-IV

Office Automation Tools Word Processing: Editing features, formatting features, saving, printing, table handling, page settings, spell-checking, macros, mail-merge, and equation editors.
Excel/Access Power Point Slides: Templates, views, formatting slide, slides with graphs, animation, using special features, presenting slide shows.

Suggested Readings:

1. Norton, P., *Introduction to Computers*, Mc-Graw-Hill, 2017.
2. Raja, Raman V., *Fundamental of Computers*, Prentice Hall of India, 2014.
3. Sanders, D. H., *Computer Today*, Mc-Graw Hill, 1988.
4. Sinha, P.K. and Sinha, P., *Computer fundamentals*, BPB Publications, 2010.
5. Vermaat, M.E., *Discovering Computers & Microsoft Office 2013: A Fundamental Combined Approach*, Cengage Learning, 2013.



Internet Fundamentals
Course Code: SBS CS 01 01 02 E 3104

L T P **Total Credits: 0(Bridge Course)/ 4(GEC)**
3 1 0

Course Objectives:
This course aims to introduce the building blocks of the internet and to provide the necessary skills to utilize the internet efficiently.

Course Level Learning Outcomes:
Upon successful completion of the course students will be able to:

- Describe how the Internet works.
- Summarize describe connections that need to be made in order to access the internet.
- Demonstrate internet tools technologies including current web-based applications, e-mail, and social networking tools.
- List the privacy & security protocols involved in the internet communication.

UNIT-I

Electronic Mail: Introduction, advantages and disadvantages, Userids, Pass words, e-mail addresses, message components, message composition, mailer features, E-mail inner workings, E-mail management, Mime types, Newsgroups, mailing lists, chat rooms.

UNIT-II

The Internet: Introduction to networks and internet, history, Working of Internet, Internet Congestion, Modes of Connecting to Internet, Internet Service Providers(ISPs), Internet addressing, comparison of IPv4 and IPv6.

UNIT-III

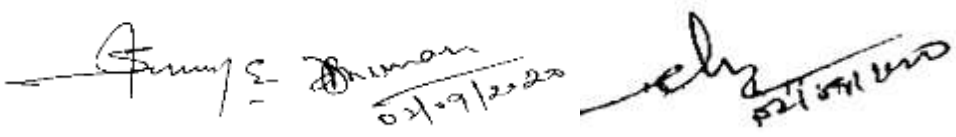
Languages and Servers: Basic and advanced HTML, XML basics. Introduction to Web Servers: PWS, IIS, Apache; Microsoft Personal Web Server. Accessing & using these servers.

UNIT-IV

Privacy and Security Topics: Introduction, Encryption schemes, Secure Web document, Digital Signatures, Firewalls.

Suggested Readings:

1. Castro, E., *HTML for the World Wide Web with XHTML and CSS: Visual QuickStart Guide*, Peachpit Press, 2006.
2. Comer, D.E. and Droms, R.E., *Computer Networks and Internets,* Prentice-Hall, Inc., 2003.
3. Deitel, H.M., Deitel, P.J. and Nieto, T.R., *Internet & World Wide Web How to Program*, Pearson Education, 2011.
4. Gralla, P., *How the Internet Works*, QUE Publication, 2006.
5. Greenlaw, R. and Hepp, E., *Inline/Online: Fundamentals of the Internet and The World Wide Web*, McGraw-Hill Higher Education, 2001.



Computer Programming using C
Course Code: SBS CS 01 01 03 E 3104

L T P **Total Credits: 0(Bridge Course)/ 4(GEC)**
3 1 0

Course Objectives:
The course is designed to provide knowledge of C language. Students will be able to develop logics which will help them to create programs, applications in C. Also by learning the basic programming constructs they can easily switch over to any other language in future. Student will learn the fundamental programming concepts and methodologies

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Design programs connecting decision structures, loops and functions.
- Explain the difference between call by value and call by address.
- Understand the dynamic behaviour of memory by the use of pointers.
- Understand the arrays and difference between structure and union.

UNIT-I

Elements of C: character set identifier and keywords, data type, declaration and definition. Operators: arithmetic, relational, logical, bit wise, unary, assignment and conditional operators their hierarchy arid associativity.

UNIT-II

Control statements: sequencing, selection, if and switch statement; repetition / loop statements: for, while, and do while loops; break, continue and goto statements.

UNIT-III

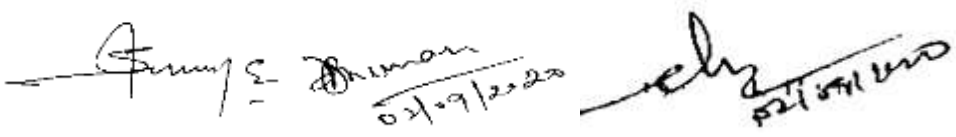
Function: definition, declaration, and Calling, call by value, call by reference prototype, passing parameters, actual and formal parameters, recursion.

UNIT-IV

Data Structures: arrays, structure, structure members, access to structure members union, string, data files. Pointer: declaration, operation of pointers, array to pointers, pointers to arrays.

Suggested Readings:

1. Gottfried, B.S., *Programming with C*., McGraw Hill Education, 2018.
2. Hanly, J. R., Koffman, E.B. , *Problem Solving and Program Design in C*, 8th edition., Pearson Publications, 2015.
3. Kanetkar, Y., *Let Us C*, 16th Edition, BPB Publication, 2017.
4. Kelley, A., Pohl, I., *A Book on C: Programming in C*, Addison Wesley, 2000.
5. Kernighan, B.W. and Ritchie D., *The C Programming Language,* Pearson Publications, 2015.



Computer Programming using C -Lab-I
Course Code: SBS CS 01 01 04 E 0021

L T P **Total Credit: 0**
0 0 2

Course Objectives:
The major objective of this course is to provide the basic knowledge of C language with the understanding of code organization, data types and functional hierarchy. Students will understand a defensive programming concept. Ability to handle possible errors during program execution.

Course Level Learning Outcomes:
Upon successful completion of the course students will be able to:

- Understand the need of programing language
- Use various data types as per requirements
- Design programs connecting decision structures, loops and functions.
- Explain the difference between call by value and call by address.
- Understand the dynamic behaviour of memory by the use of pointers.
- Understand the arrays and difference between structure and union
- Work with textual information, characters and strings.
- Work with arrays of complex objects.
- Understanding a concept of object thinking within the framework of functional model.

Data Structures
Course Code: SBS CS 01 01 01 C 4004

L T P **Total Credits: 4**
4 0 0

Course Objectives:

Using computer science theory, students will construct and analyze various data structures and abstract data types including lists, stacks, queues, trees, and graphs. Students will implement various sorting, searching, and hashing algorithms. Students will build a substantial, complex data structure.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Design correct programs to solve problems.
- Choose efficient data structures and apply them to solve problems.
- Analyze the efficiency of programs based on time complexity.
- Prove the correctness of a program using loop invariants, preconditions and post conditions in programs.

UNIT-I

Introduction: Basic Terminology, Elementary Data Organization, Structure Operations, Algorithm, Complexity and Time-Space trade-off.

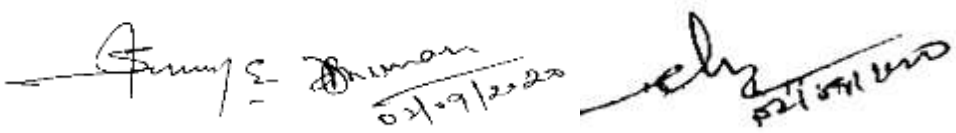
Arrays: Array Definition, Representation and Analysis, Single and Multidimensional Arrays, Address Calculation, Application of Arrays.

UNIT-II

Stacks: Array Representation and Implementation of the stack, Operations on Stacks: Push & Pop, Array Representation of Stack, Linked Representation of Stack, Operations Associated with Stacks, Application of stack: Conversion of Infix to Prefix and Postfix Expressions, Evaluation of postfix expression using stack.

Queues: Array and linked representation and implementation of queues, Operations on Queue: Create, Add, Delete, Full and Empty, Circular queues, D-queues and Priority Queues.

Linked list: Representation and Implementation of Singly Linked Lists, Traversing and Searching of Linked List, Overflow and Underflow, Insertion and deletion to/from Linked Lists, Insertion and deletion Algorithms, Doubly Linked List, Linked List in Array.



UNIT-III

Trees: Basic terminology, Binary Trees, Binary tree representation, Array and Linked Representation of Binary trees, Types of Binary Tree, Traversing Binary trees, Binary Search Tree (BST), Insertion and Deletion in BST, AVL Trees, Huffman algorithm.

Graphs: Terminology & Representations, Graphs & Multi-graphs, Directed Graphs, Sequential Representations of Graphs, Adjacency Matrices, Traversal, Connected Component and Spanning Trees, Minimum Cost Spanning Trees.

UNIT-IV

Searching and Hashing: Sequential search, binary search, comparison and analysis, Hash Table, Hash Functions, Collision Resolution Strategies, Hash Table Implementation.

Sorting: Insertion Sort, Bubble Sorting, Selection Sort, Quick Sort, Merge Sort, Heap Sort, Linear time sorting, Practical consideration for Internal Sorting and External Sorting.

Suggested Readings:

1. Cormen, T.H., Leiserson, C.E., Rivest, R.L. and Stein, C., *Introduction to Algorithms,* MIT Press, 2010.
2. Goodrich, M.T., Tamassia, R. and Mount, D.M., *Data Structures and Algorithms in C++*, John Wiley & Sons, 2016.
3. Langsam, Y., Augenstein, M. and Tenenbaum, A.M., *Data Structures using C and C++,* Prentice Hall, 2015.
4. Lipschutz, S., *Schaum's Outline of Theory and Problems of Data Structures*, McGraw-Hill, 2014
5. Thareja, R., *Data Structures using C*, Oxford University Press, 2014.

Computer Networks
Course Code: SBS CS 01 01 02 C 4004

L T P **Total Credits: 4**
4 0 0

Course Objectives:
The main emphasis of this course is on the organization and management of local area networks (LANs). The course objectives include learning about computer network organization and obtaining a theoretical understanding of data communication and computer networks.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Analyze the requirements for a given organizational structure and select the most appropriate networking architecture and technologies.
- Discuss and identify deficiencies in existing protocols, and then go onto formulating new and better protocols.
- Analyze, specify and design the topological and routing strategies for an IP based networking infrastructure.
- Describe a working knowledge of datagram and internet socket programming.

UNIT-I

Introduction To Computer Networks: Definition of a Computer Network, The OSI Reference Model, The TCP/IP Reference Model, Protocols and Hardware involved in the OSI model, Comparison of the OSI & the TCP/IP.

Application Layer: Domain name space, DNS in internet, electronic mail, FTP, WWW, HTTP, SNMP, multimedia, network security

UNIT-II

Physical Layer: Introduction: Network topologies; Linear Bus Topology, Ring Topology, Star Topology, Hierarchical or Tree Topology, Topology Comparison, Considerations when choosing a Topology: Switching; Circuit switching, Message switching, Packet switching.

Transmission Medium: Introduction: Transmission medium; Guided & Unguided Transmission medium, Twisted pair, Coaxial cable, Optical fiber, Comparison of fiber

optics and copper wire: Wireless transmission; Electromagnetic spectrum, Radio transmission, Microwave transmission.

UNIT-III

Data Link Layer: Introduction; Goal of DLL: Design issues of DLL; Services provided to the Network layer, Framing, Error control, Flow control, ARQ strategies: Stop-and-Wait, RTT estimation, sliding window, Go-Back-N retransmission, Error Detection and correction: Parity bits, Single bit error correction or (n, m), Error Detection or Cyclic Redundant Code (CRC): Data Link layer protocols; Transmission control protocols, HDLC.

UNIT-IV

Network Layer: Introduction: Design issues of Network layer; Nature of the service provided, Internetworking: Principles of Routing; Types of routing algorithms, Properties of routing algorithms, Optimality principle: Routing algorithms; Shortest path algorithm, Flooding, Distance vector routing, Hierarchical routing, Link state routing, Congestion: Factors of congestion, Comparison of flow control and congestion control, General principles of congestion control, Closed loop solution: IP protocol (IPV4).

Transport Layer: Introduction: Services of Transport layer; Service primitives: Connection establishment: Connection Release: Transport Protocols; TCP protocol, UDP protocol

Suggested Readings:

1. Comer, D.E., and Droms, R.E., *Computer Networks and Internets,* Prentice-Hall Inc., 2018.
2. Forouzan, A.B., *Data Communications & Networking,* Tata McGraw-Hill Education, 2017.
3. Kundu, S., *Fundamentals of Computer Networks*, PHI Learning Pvt. Ltd., 2008.
4. Kurose, J.F., *Computer Networking: A Top-Down Approach Featuring the Internet,* Pearson Education India. 2016.
5. Stallings, W.S., *Data and Computer Communications,* Pearson Education India, 2013.

Computer Organization and Architecture

Course Code: SBS CS 01 01 03 C 4004

L T P **Total Credits: 4**
4 0 0

Course Objectives:

This course is intended to teach the basics involved in data representation and digital logic circuits used in the computer system. This includes the general concepts in digital logic design, including logic elements, and their use in combinational and sequential logic circuit design. This course will also expose students to the basic architecture of processing, memory and i/o organization in a computer system.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Describe the structure, function and characteristics of computer systems.
- Outline the design of the various functional units and components of computers.
- Define the function of each element of a memory hierarchy,
- Identify and compare different methods for computer I/O.

UNIT-I

Information Representation: Number systems, BCD codes, character codes, error detecting and correcting codes, fixed-point and floating-point representation of information. Binary arithmetic operations, Booths multiplication. Binary Logic: Boolean algebra, Boolean functions, Truth Tables, Canonical and Standard forms, Simplification of Boolean functions, Digital logic gates.

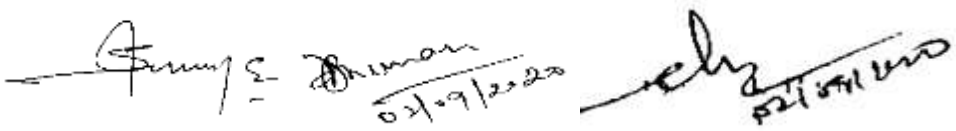
UNIT-II

Combinational Logic: Design procedure, Adders, Subtractors, Encoders, Decoders, Multiplexers, De-multiplexers and Comparators.

Sequential Logic: Latches, Flip-Flops, Shift Registers and Counters.

UNIT-III

Memory System: Characteristics of memory system, Memory hierarchy, Cache Memory- Cache memory principles, Elements of cache design- cache address, size, mapping functions, replacement algorithms, write policy, Internal Memory- semiconductor memory, External Memory- Hard Disk organization.



UNIT-IV

CPU Organizations: Processor organization, machine instructions, instruction cycles, instruction formats and addressing modes, microprogramming concepts, micro-program sequencer. I/O organization: I/O interface, interrupt structure, transfer of information between CPU/memory and I/O devices and IOPs. Case Study: Von-Neumann Architecture

Suggested Readings:

1. Hamacher, V.C., Vranesic, Z.G., Zaky, S.G., Vransic, Z. and Zakay, S., *Computer Organization,* McGraw-Hill, 2011.
2. Hayes, J.P., *Computer Architecture and Organization,* McGraw-Hill Inc., 2017.
3. Mano, M.M., *Computer System Architecture,* Prentice-Hall of India, 2017.
4. Sarangi, S.R., *Computer Organisation and Architecture,* McGraw-Hill, 2017.
5. Stallings, W., *Computer Organization and Architecture: Designing for Performance*, Pearson Education India, 2016.

Discrete Mathematical Structures

Course Code: SBS CS 01 01 04 C 4004

L T P **Total Credits: 4**
4 0 0

Course Objectives:

The objective of this course is to teach students how to think logically and mathematically. The course stresses on mathematical reasoning and describes different ways in which mathematical problems could be solved. There are two thematic areas covered in this course: mathematical reasoning and discrete structures.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Describe the notion of mathematical thinking, mathematical proofs, and algorithmic thinking, and be able to apply them in problem solving.
- Apply the basics of discrete probability and number theory in problem solving.
- Apply algebraic techniques to analyse basic discrete structures and algorithms.
- Relate the properties of graphs and related discrete structures to practical examples.

UNIT-I

Set Theory: Definition of Sets, Venn Diagrams, complements, Cartesian products, power sets, counting principle, cardinality and countability (Countable and Uncountable sets), proofs of some general identities on sets, pigeonhole principle.

Relation: Definition, Types of relation, the composition of relations, domain and range of a relation, pictorial representation of a relation, properties of relation, partial ordering relation, Lattices, Hasse diagram.

UNIT-II

Algebraic Structure: Binary composition and its properties definition of algebraic structure.

Groups: Semi-group, Monoid Groups, Abelian group, properties of groups, Permutation Groups, Sub Group, Cyclic Group, Rings and Fields (definition and standard results).

UNIT-III

Propositional Logic: Proposition logic, basic logic, logical connectives, truth tables, tautologies, contradiction, normal forms (conjunctive and disjunctive), modus Ponens and modus Tollens, validity, predicate logic, universal and existential quantification, Boolean expressions, Karnaugh map.

UNIT-IV

Graphs: Graph terminology, types of graph, connected graphs, components of the graph, Euler graph, Hamiltonian path and circuits, Graph coloring, Chromatic number.

Tree: Definition, Types of trees(rooted, binary), properties of trees, binary search tree, tree traversing (preorder, inorder, postorder).

Suggested Readings:

1. Kolman, B., Busby, R.C., and Ross, S.C., *Discrete Mathematical Structures*, Prentice-Hall, 2008.
2. Lipschutz, S., and Lipson, M.L., *Discrete Mathematics*, McGraw-Hill, 2017.
3. Liu, C.L., and Mohapatra, D.P., *Elements of Discrete Mathematics: A Computer Oriented Approach,* Tata McGraw-Hill, 2017.
4. Rosen, K.H., and Krithivasan, K., *Discrete Mathematics and Its Applications: With Combinatorics and Graph Theory*, Tata McGraw-Hill Education, 2017.
5. Sarkar, S.K., *A Textbook of Discrete Mathematics*, S. Chand Publishing, 2016.

Operating System and Shell Programming

Course Code: SBS CS 01 01 05 C 4004

L T P **Total Credits: 4**
4 0 0

Course Objectives:

A successful student will be able to understand the basic components of a computer operating system, and the interactions among the various components. The course will cover an introduction on the policies for scheduling, deadlocks, memory management, synchronization, system calls, and file systems. The students will implement solutions via C/C++ programs.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Demonstrate the important Linux library functions and system calls.
- Describe the inner workings of Linux operating systems.
- Write shell scripts to perform repetitive tasks using while and for loops.
- Design and implement shell functions.

UNIT-I

Operating System Introduction: function, characteristics, structures–simple batch, multiprogrammed, timeshared, personal computer, parallel, distributed systems, real-time systems, system components, operating system services, system calls, virtual machines.
Process and CPU Scheduling: Process concepts and scheduling, operation on processes, cooperating processes, threads and inter-process communication scheduling criteria, scheduling algorithm, multiple-processor scheduling, real time scheduling.

UNIT-II

Management and Virtual memory: logical versus physical address space, swapping, contiguous allocation, paging, segmentation, segmentation with paging. Demand paging, performance of denuding paging, page replacement, page replacement algorithm, allocation of frames, thrashing.

UNIT-III

File System Interface and Implementation: access methods, directory, structure, protection, file system structure, allocation methods, free space management, directory management, directory implementation, efficiency and performance.
I/O Management: I/O software and its types, disk scheduling.

Process Management and Synchronization: Critical section problem, synchronization, critical regions, monitors.

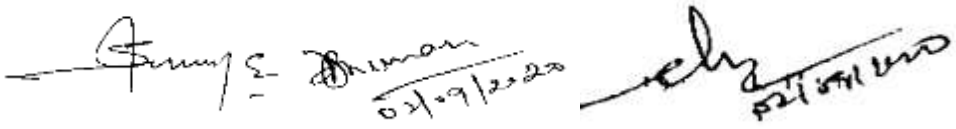
UNIT-IV

Deadlocks: system model, dead locks characterization, methods for handling deadlocks, deadlock prevention, deadlock avoidance, deadlock detection and recovery from deadlock.

Shell Programming: vi editor, shell variables, I/O in shell, control structures, loops, subprograms, creating shell scripts. Basic system administration in Linux/Unix.

Suggested Readings:

1. Das, S., *Your UNIX: The Ultimate Guide*, McGraw-Hill Inc., 2012.
2. Goerzen, J., *Linux Programming Bible*, IDG, 2000.
3. Kanetkar, Y.P., *UNIX Shell Programming*, BPB Publications, 2003.
4. Prata, S., and Waite Group, *Advanced UNIX: A Programmer's Guide*, HW Sams, 1985.
5. Venkateshmurthy, M.G., *Introduction to Unix and Shell Programming,* Pearson Education India, 2009.



Database Management System
Course Code: SBS CS 01 02 09 C 4004

L T P **Total Credits: 4**

4 0 0

Course Objectives:

This course is intended to provide an introduction to the management of database systems. The course emphasizes the understanding of the fundamentals of relational systems including data models, database architectures, and database manipulations. The course uses a problem-based approach to learning.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Elaborate on different issues involved in the design and implementation of a database system. Study the physical and logical database designs, database modeling, relational, hierarchical, and network models.
- Analysis and Practice on data manipulation language to query, update and manage a database.
- Determine essential DBMS concepts such as database security, integrity, concurrency, distributed database, and intelligent database, Client/Server (Database Server), Data Warehousing.
- Develop a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.

UNIT-I

Basic Concepts: File Systems vs. DBMS, Characteristics of the Database Approach, Abstraction and Data Integration, Database users, Advantages and Disadvantages of a DBMS.

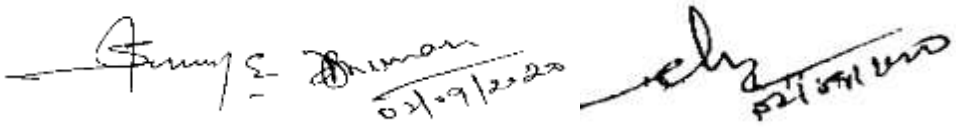
Database Systems Concepts and Architecture: Data Models, Schema and Instances, DBMS architecture and Data Independence, Database languages and Interfaces, DBMS functions and component modules.

UNIT-II

Entity-Relationship Model: Entity Types, Entity Sets, Attributes & keys, Relationships, Relationships Types, Roles and Structural Constraints, Design issues, E-R Diagrams, Design of an E-R Database Schema, Reduction of an E-R Schema to Tables.

Relational Data Model: Relational model concepts, Integrity constraints over Relations, Relational Algebra – Basic Operations.

SQL: DDL, DML, and DCL, views & Queries in SQL, Specifying Constraints & Indexes in SQL.



UNIT-III

Relational Database Design: Functional Dependencies, Decomposition, Normal forms based on primary keys (1 NF, 2 NF, 3 NF, & BCNF)
Transaction Processing Concepts: Introduction to Transaction Processing, Transaction & System Concepts, Properties of Transaction, Schedules and Recoverability, Serializability of Schedules.
Concurrency Control Techniques: Locking Techniques, Timestamp ordering, Multi-version Techniques, Optimistic Techniques, Granularity of Data items.

UNIT-IV

Databases for Advanced Applications: Active database concepts, Temporal database concepts, Spatial databases, Deductive databases; Emerging Database Technologies: Mobile databases, Multimedia Databases, Geographic information systems (GIS); XML and Internet Databases: Structured, Semi-structured and Unstructured Data, Introduction to web databases and XML, Structure of XML data.

Suggested Readings:

1. Bayross, I., *SQL, PL/SQL: The Programming Language of Oracle,* BPB Publications. 2010.
2. Connolly, T.M. and Begg, C.E., *Database Systems: A Practical Approach to Design, Implementation, and Management*, Pearson Education, 2019.
3. Date, C.J., *An Introduction to Database Systems*, Pearson Education India, 2012.
4. Elmasri, R., *Fundamentals of Database Systems*, Pearson Education India, 2015.
5. Silberschatz, A., Korth, H.F. and Sudarshan, S., *Database System Concepts,* McGraw-Hill, 2013.

Computer Graphics
Course Code: SBS CS 01 02 10 C 4004

L T P **Total Credits: 4**
4 0 0

Course Objectives:

The main objective of this course is to introduce the concepts of computer graphics to the students. It starts with an overview of interactive computer graphics, two-dimensional system and mapping, then it presents the most important drawing algorithm, two-dimensional transformation; Clipping, filling and an introduction to 3-D graphics.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Describe the basic concepts of computer graphics.
- Implement various scan conversion problems with programming.
- Demonstrate various transformations on the digital drawings.
- Apply various modelling techniques in multimedia object creation.

UNIT-I

Introduction: Survey of computer Graphics and its applications; Interactive and passive graphics; display processors; Graphic Devices: Display systems-refresh CRTs, raster scan and random scan monitors, Grey shades, Interlacing, beam penetration shadow mask monitors, lookup tables, plasma panel, LED and LCD monitors, VGA and SVGA resolutions; Hard copy Devices-printers, plotters; Interactive Input Devices.

UNIT-II

Drawing Geometry: Coordinate system; resolution; use of the homogeneous coordinate system; scan conversion: symmetrical DDA, simple DDA, Bresenham's line drawing algorithm, Circle drawing using DDA and polar coordinates, Bresenham's circle drawing algorithm, generation of an ellipse. Curve Drawing

UNIT-III

2-D Transformations: Translation; rotation; scaling; mirror reflection; shearing; zooming; panning; input techniques-pointing, positioning, rubber band methods and dragging; tweening, Morphing. Graphic operations: Clipping-line clipping using Sutherland-Cohen and midpoint sub- division algorithm, Liang Barsky Line clippers algorithm, polygon clipping; window and viewport; windowing transformation; Filling algorithms.

UNIT-IV

3-D Graphics: 3D modelling of objects; 3D display techniques; coordinate system; 3D transformation matrices for translation, scaling and rotation; parallel projection; perspective projection; Hidden-surface removal - Z-buffer, back face, scan-line, depth-sorting, area subdivision; Shading - modelling light intensities, Gouraud shading, Phong shading.

Suggested Readings:

1. Enderle, G., Kansy, K. and Pfaff, G., *Computer Graphics Programming: GKS—The Graphics Standard,* Springer Science & Business Media. 2012.
2. Hearn, D., Baker, M.P. and Carithers, W.R., *Computer Graphics with OpenGL*, Prentice Hall, 2014.
3. Foley, J.D., Van, F.D., Van Dam, A., Feiner, S.K., Hughes, J.F., Angel, E. and Hughes, J., *Computer Graphics: Principles and Practice,* Addison-Wesley Professional. 2018.
4. Klinger, A., Fu, K.S. and Kunii, T.L., *Data Structures, Computer Graphics, and Pattern Recognition*, Academic Press. 2014.
5. Marschner, S. and Shirley, P., *Fundamentals of Computer Graphics,* CRC Press, 2015.

Object Oriented Programming
Course Code: SBS CS 01 02 11 C 4004

L T P **Total Credits: 4**
4 0 0

Course Objectives:

The objective of this course is to develop programming skills of students, using object-oriented programming concepts, learn the concept of class and object using C++ and develop classes for simple applications.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Analyze fundamentals of programming such as variables, conditional and iterative execution, methods, etc.
- Discuss the fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries, etc.
- Summarize important topics and principles of software development.
- Practice on writing a computer program to solve specific problems.

UNIT-I

Object-Oriented Concepts: Data abstraction, Data Hiding, Encapsulation, polymorphism, modularity, hierarchy, typing, concurrency, persistence. C++ Basics: Classes and Objects, Data types, loops and decisions, structures and functions, Scope of class and its member, Nested Class, object arrays, Pointers, Constructor: parameterized constructor, multiple constructors, default constructor, copy constructor, implicit constructor, destructor function, dynamic allocation operators: new(), delete().

UNIT-II

Inheritance: Base and Derived Classes, Single inheritance, Multilevel inheritance, Hierarchical inheritance, Hybrid Inheritance, Multiple inheritance, Protected Members, Casting Base-Class Pointers to Derived- Class Pointers, Using Member Functions, Overriding Base–Class Members in a Derived Class, Public, Protected and Private Inheritance, Using Constructors and Destructors in derived Classes.

UNIT-III

Polymorphism: Compile time and Run time, Abstract class, Virtual class, Virtual base classes, pointers to base and derived classes, virtual functions, early and late binding, Pure virtual function, virtual destructor, virtual derivation. Friend function & Friend class, Inline functions, function overloading, Operator Overloading: Unary, Binary.

UNIT-IV

Generic Programming: Function Templates, Overloading Template Functions, Class Template, Class Templates and Non-Type Parameters.

Exception Handling: Try, Throw, Catch, Throwing an Exception, Catching an Exception, Re-throwing an Exception.

File Handling: Hierarchy of File Stream classes, Opening and Closing files, File modes, testing for errors, File pointers and their manipulations, ASCII & Binary files, Sequential and Random access files. files, opening & closing a file, read() & write() functions, File manipulation using seekg(), tellg() functions.

Suggested Readings:

1. Balaguruswami, E., *Object-Oriented Programming in C++*, McGraw Hill Education, 2017.
2. Kanetkar, Y.P., *Let us C++*, BPB Publications, 2019.
3. Lafore, R., *Object-oriented programming in C++*, Pearson Education, 2001.
4. Mothe, M., *C++ Programming: A Practical Approach,* Pearson Education India, 2011.
5. Schildt, H., *C++ The Complete Reference*, McGraw-Hill Osborne Media, 2014.
6. Stroustrup, B., *The C++ Programming Language*, Pearson Education India, 2013.

Software Engineering
Course Code: SBS CS 01 02 12 C 4004

L T P **Total Credits: 4**
4 0 0

Course Objectives:

The objective of this course is to provide a solid fundamental knowledge of software engineering. This course will help the students to utilize and exhibit strong communication and interpersonal skills, as well as professional and ethical principles when functioning as members and leaders of multi-disciplinary teams.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Interpret the problem statement for the software design.
- Analyze and specify the requirements of the software efficiently.
- Translate the requirements into the design model with modern tools.
- Write the test cases and analyze the software modules.

UNIT-I

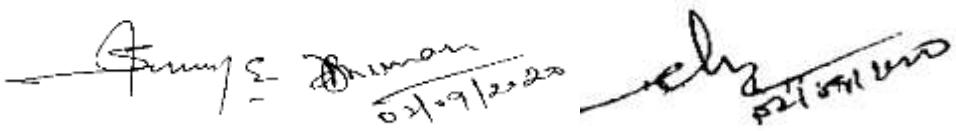
Software and Software Engineering: software characteristics, software crisis, software engineering paradigms. Planning a Software Project: software cost estimation, project scheduling, personal planning, team structure

UNIT-II

Software Configuration Management: quality assurance, project monitoring, risk management. Software Requirement Analysis: structured analysis, object-oriented analysis and data modeling, software requirement specification, validation.

UNIT-III

Design and Implementation of Software: software design fundamentals, design methodology (structured design and object-oriented design), design verification, monitoring and control, coding.
Software Reliability: metric and specification, fault avoidance and tolerance, exception handling, defensive programming.



UNIT-IV

Testing: testing fundamentals, white box and black box testing, software testing strategies; unit testing, integration testing, validation testing, system testing, debugging Software

Maintenance: maintenance characteristics, maintainability, maintenance side effects, CASE tools

Suggested Readings:

1. Aggarwal, K.K. and Singh, Y., *Software Engineering*, New Age International, 2001.
2. Fairley, R., *Software Engineering Concepts,* McGraw-Hill, 2017.
3. Jalote, P., *An Integrated Approach to Software Engineering,* Springer Science & Business Media, 2012.
4. Pressman, R.S., *Software Engineering: A Beginner's Guide,* McGraw-Hill, 2019.
5. Sommerville, I., *Software Engineering*, Pearson Education India, 2004.

Design and Analysis of Algorithms

Course Code: SBS CS 01 02 13 C 4004

L T P **Total Credits: 4**
4 0 0

Course Objectives:

The objective of this course is to reinforce basic design concepts (e.g., pseudo code, specifications, top-down design) and have the knowledge of algorithm design strategies. This course emphasizes mainly on the analysis of an algorithm w.r.t. time and space complexity.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Analyze the asymptotic performance of algorithms.
- Write rigorous correctness proofs for algorithms.
- Demonstrate familiarity with major algorithms and data structures.
- Relate the real-life problems and their better solution.

UNIT-I

Introduction to analysis of algorithms: Analysis of algorithms, asymptotic notation-Big-O, Omega and Theta notations, recurrence relations, solving recurrences, Abstract data types, Linear Data Structures and their sequential storage representation: stacks, queues, priority queues, and their applications.

UNIT-II

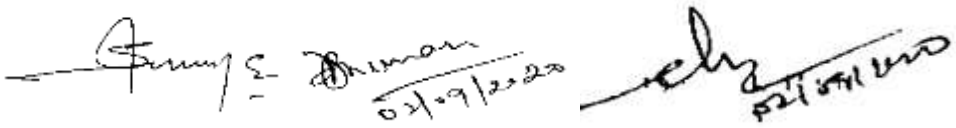
Divide and Conquer: General method, Binary Search, Exponentiation problem, Merge Sort, Quick Sort, Selection Sort, Strassen's Matrix Multiplication algorithms and analysis of algorithms for these problems.

Greedy Method: General method, Knapsack Problem, Job sequencing with deadlines, Minimum Spanning trees, Single source shortest path and analysis of these algorithms.

UNIT-III

Dynamic Programming: General method, 0/1 Knapsack problem, Optimal BST, All Pairs shortest path, Traveling Salesman Problem, Longest common subsequence(LCS).

Back Tracking: General method, 8 queen's problem, graph coloring, hamiltonian cycles and analysis of these problems.

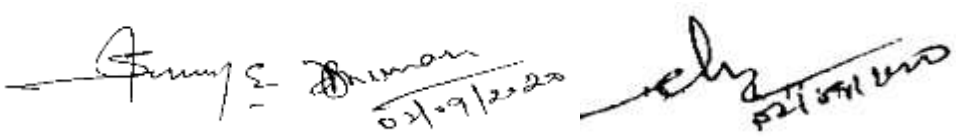


UNIT-IV

NP-Hard and NP-Complete Problems: P, NP, NP-Hard & NP-Complete Classes, Reductions: Vertex cover, Simple Max-Cut, Hamiltonian Circuit, Traveling salesman problem, kernel, 3- dimensional matching, and other NP-Complete Problems, Satisfiability and variations, Cook's theorem, examples of NP-Hard problems.

Suggested Readings:

1. Aho, A.V. and Hopcroft, J.E., *The Design and Analysis of Computer Algorithms,* Pearson Education India, 2002.
2. Cormen, T.H., Leiserson, C.E., Rivest, R.L. and Stein, C., *Introduction to Algorithms,* MIT Press. 2010.
3. Horowitz, E. and Sahni, S., *Fundamentals of Computer Algorithms,* Computer Science Press, 2008.
4. Kleinberg, J. and Tardos, E., *Algorithm Design,* Pearson Education India, 2013.
5. Sridhar, S., *Design and Analysis of Algorithms*, Oxford University Press, 2015.



Mobile Communication
Course Code: SBS CS 01 02 05 E 3003

L T P **Total Credits: 3**
3 0 0

Course Objectives:

This course will provide students with both broad and in-depth knowledge, and a critical understanding of mobile computing from different viewpoints: infrastructures, principles and theories, technologies, and applications in different domains. Student will understand the operation of mobile communications systems and their generation divisions.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Describe the fundamentals of wireless communications.
- Analyze security, energy efficiency, mobility, scalability, and their unique characteristics in wireless networks.
- Demonstrate basic skills for cellular networks design.
- Apply knowledge of TCP/IP extensions for mobile and wireless networking.

UNIT-I

Introduction to Mobile Communications and Computing: Mobile Computing (MC): Introduction to MC, novel applications, limitations, and architecture.

GSM: Mobile services, System architecture, Radio interface, Protocols, Localization and calling, Handover, Security, and New data services.

(Wireless) Medium Access Control: Motivation for a specialized MAC (Hidden and exposed terminals, Near and far terminals), SDMA, FDMA, TDMA, CDMA.

UNIT-II

Mobile Network Layer: Mobile IP (Goals, assumptions, entities and terminology, IP packet delivery, agent advertisement and discovery, registration, tunneling and encapsulation, optimizations), Dynamic Host Configuration Protocol (DHCP).

Mobile Transport Layer: Traditional TCP, Indirect TCP, Snooping TCP, Mobile TCP, Fast retransmit/fast recovery, Transmission /time-out freezing, Selective retransmission, Transaction oriented TCP.

UNIT-III

Database Issues: Hoarding techniques, caching invalidation mechanisms, client server computing with adaptation, power-aware and context-aware computing, transactional models, query processing, recovery, and quality of service issues. Data Dissemination: Communications asymmetry, classification of new data delivery mechanisms, push based mechanisms, pull-based mechanisms, hybrid mechanisms, selective tuning (indexing) techniques.

UNIT-IV

Mobile Ad hoc Networks (MANETs): Overview, Properties of a MANET, spectrum of MANET applications, routing and various routing algorithms, security in MANETs. **Protocols and Tools:** Wireless Application Protocol-WAP. (Introduction, protocol architecture, and treatment of protocols of all layers), Bluetooth (User scenarios, physical layer, MAC layer, networking, security, link management) and J2ME.

Suggested Readings:

1. DasBit, S. and Sikdar, B.K., *Mobile Computing*, PHI Learning Pvt. Ltd., 2009.
2. Kamal, R., *Mobile Computing,* Oxford University Press, 2008.
3. Schiller, J.H., *Mobile Communications,* Pearson Education, 2004.
4. Stojmenovic, I., *Handbook of Wireless Networks and Mobile Computing,* Wiley, 2002.
5. Talukdar, A.K., *Mobile Computing,* Tata McGraw-Hill Education, 2010.

Management of Information System and E-Commerce
Course Code: SBS CS 01 02 06 E 3003

L T P **Total Credits: 3**
3 0 0

Course Objectives:

This course focuses on principles of e-commerce from a business perspective, providing an overview of business and technology topics, business models, virtual value chains and social innovation and marketing strategies. In addition, some of the major issues associated with e- commerce security, privacy, intellectual property rights, authentication, encryption, acceptable use policies, and legal liabilities will be explored. Students will build their own web presence and market it using an online platform.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Identify the relationship between the digital firm, electronic commerce, electronic business and internet technology.
- Analyze the relationship between organizations, information systems and business processes, including the processes for customer relationship management and supply chain management.
- Demonstrate an understanding of the foundations and importance of E-commerce
- Estimate the effect of changing technology on traditional business models and strategy;

UNIT-I

Introduction to the MIS concept- Definition, Role of the MIS, Impact of MIS, MIS and the user, Management as a control system, MIS support to the management, Management effectiveness and MIS, Organization as a system. MIS: organization effectiveness.

Decision Making and DSS- Decision making concepts, decision-making process, decision- making by analytical modelling, Behavioral concepts in decision making, organizational decision-making, Decision structure, DSS components, Management reporting alternatives.

UNIT-II

Enterprise Business system- Introduction, cross-functional enterprise applications, real-world case, Functional business system, Introduction, marketing systems, sales force automation, CIM, HRM, Customer relationship management, ERP, Supply chain management.

Client-Server Architecture and E-business Technology- Client-server architecture,

implementation strategies, Introduction to E-business, the model of E-business, Internet and World Wide Web, Intranet/Extranet, Electronic, Impact of Web on Strategic management, MIS in Web environment.

UNIT-III

Introduction to e-commerce, E-commerce Business Models and Concepts, Ecommerce Infrastructure: The Internet and World Wide Web, Web design, JavaScript Internet Information Server (IIS); Personal Web Server (PWS),

E-Commerce techniques and Issues- Introduction to Active Server Pages (ASP), Building an E-Commerce Web Site, E-Commerce Payment Systems, E-Commerce Marketing Techniques, Building product catalogue, Search Product catalogue, Web Spider and search agent, Ethical, Social and Political Issues in ECommerce.

UNIT-IV

Internet Communication- Transaction Systems, Shopping Carts, XML, E-Commerce Applications: Business-to-Consumer(B2C), Consumer-to-Consumer (C2C), Business- to-Business (B2B), Digital Government, Marketplaces, and Communities, Security and Encryption, Web Security.

Suggested Readings:

1. Jawadekar, W.S., *Management Information Systems: Text and Cases: A Global Digital Enterprise Perspective*, Tata McGraw-Hill Education, 2013.
2. Kalakota, R. and Whinston, A.B., *Electronic Commerce: A Manager's Guide,* Addison-Wesley Professional, 1997.
3. Laudon, K.C. and Laudon, J.P., *Management Information Systems,* Pearson, 2018.
4. Laudon, K.C. and Traver, C.G., *E-Commerce: Business, Technology, Society,* Pearson, 2016.
5. Rayport, J.F. and Jaworski, B.J., *Introduction to E-Commerce,* McGraw-Hill, 2003.

Quantum Computing
Course Code: SBS CS 01 02 07 E 3003

L T P **Total Credits: 3**
3 0 0

Course Objectives:

This course aims to introduce the fundamentals of quantum computation. This course provides an interdisciplinary introduction to the emerging field of quantum computer science, explaining basic quantum mechanics, quantum entanglement, its structure and its physical consequences and introduces quits.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Implement complex vector spaces.
- Apply quantum mechanics in quantum computing.
- Architecture the quantum algorithms.
- Learn fundamentals of quantum computation.

UNIT-I

Complex numbers and its geometrical representations, Complex vector spaces, inner products and Hilbert spaces, Hermitian and unitary matrices, Tensor products of vector spaces

UNIT-II

Deterministic Systems, Probabilistic descriptions and Quantum systems, Basics of Quantum theory, Schrodinger's time dependent equation, Wave nature of Particles, state vector, operators, postulates of quantum mechanics, Dirac formalism, Stern-Gerlach experiment, electron spin, superposition of states, entanglement

UNIT-III

Bits and Qubits, Classical gates versus quantum gates, single qubit gates, multiple qubit gates, design of quantum circuits, Deutsch's Algorithm, DeutschJozsa Algorithm, Simon's periodicity algorithm, Grover's search algorithm, Shor's Factoring algorithm

UNIT-IV

Quantum programming languages, Probabilistic and Quantum computations, introduction to quantum cryptography and quantum information theory, Comparison between classical and quantum information theory, Bell states, no cloning theorem, Quantum error correction.

Suggested Readings:

1. Griffiths, D.J. and Schroeter, D.F., *Introduction to Quantum Mechanics*, Cambridge University Press, 2018.
2. McMahon, D., *Quantum Computing Explained*, John Wiley & Sons, 2007.
3. Nielsen, M.A. and Chuang, I., *Quantum Computation and Quantum Information*, 2010.
4. Pittenger, A.O., *An Introduction to Quantum Computing Algorithms,* Springer Science & Business Media, 2012.
5. Yanofsky, N.S. and Mannucci, M.A., *Quantum Computing for Computer Scientists,* Cambridge University Press, 2008.

Theory of Computation
Course Code: SBS CS 01 02 08 E 3003

L T P **Total Credits: 3**
3 0 0

Course Objectives:

This course aims to provide a formal connection between algorithmic problem solving and the theory of languages and automata and develop them into a mathematical view towards algorithmic design and in general computation itself.

Course level learning outcome:

Upon successful completion of the course students will be able to:

- Relate practical problems to languages, automata, and computability.
- Apply mathematical and formal techniques for solving problems.
- Distinguish different computing languages and classify their respective types.
- Analyze and design finite automata, pushdown automata, Turing machines, formal languages, and grammars.

UNIT-I

Recursive Languages: Recursive definition, Alphabets, Language, Regular expression, definitions of Finite state machine, Transition graphs, Deterministic & non-deterministic finite state machines, Regular grammar, Left-linear and right linear, Thomson's construction to convert regular Expression to NDFA & subset algorithm to convert NDFA to DFA. Minimization of DFA, Finite state machine with output (Moore machine and Mealy Machine), conversion of Moore machine to Mealy machine & vice-versa.

UNIT-II

Properties of Regular Languages: Conversion of DFA to regular expression, Pumping lemma, Properties and limitations of finite state machine, Decision properties of regular languages, Application of finite automata.
Context Free Grammar: Context free grammar, Writing context free grammar for problems, Derivation tree and ambiguity, Application of context free grammars, Chomsky and Greibach Normal form, Conversion of CFG to CNF and GNF. Properties of context free grammar, CYK algorithm

UNIT-III

PDA: Push down stack machine, Design of deterministic and non-deterministic push-down stack, Parser design.
Turing Machine: Turing machine definition and design of Turing Machine, Church-

Turing Thesis, Variations of Turing Machines, combining Turing machine, Universal Turing Machine, Post Machine, Chomsky Hierarchy.

UNIT-IV

Incommutability: Halting problem, Turing enumerability, Turing acceptability and Turing decidability, Unsolvable problems about Turing machines.
Computation Complexity: P, NP and NP Complete Problems.

Suggested Readings:

1. Hopcroft, J.E., Motwani, R. and Ullman, J.D., *Introduction to Automata Theory, Languages, and Computation*, Pearson Education, 2008.
2. Kulkarni, V., *Theory of Computation*, Oxford University Press, 2013.
3. Linz, P., *An Introduction to Formal Languages and Automata*, Jones & Bartlett Learning, 2016.
4. Martin, J.C., *Introduction to Languages and the Theory of Computation,* McGraw-Hill, 2001
5. Sipser, M., *Introduction to the Theory of Computation,* Cengage Learning, 2014.

Compiler Design
Course Code: SBS CS 01 03 18 C 4004

L T P **Total Credits: 4**
4 0 0

Course Objectives:

The aim of this course is to provide students with the knowledge and abilities to design and implement compilers. It describes the steps of compilation starting with the scanner, and then, followed by the parser design and implementation. The course also provides information on semantic analysis and, local and global compiler optimization algorithms.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Apply the knowledge of lexical tool to develop a scanner & parser
- Separate the lexical, syntactic and semantic analysis into meaningful phases for a compiler to undertake language translation
- Design and develop software system for backend of the compiler
- Comprehend and adapt to new tools and technologies in compiler design

UNIT-I

Compiler structure: Analysis-Synthesis model of compilation, Various phases of a compiler, Tool based approach to Compiler Construction.

Lexical analysis: Interface with input, Parser and Symbol table, Token, Lexeme and Patterns. Difficulties in Lexical Analysis. Error reporting. Implementation. Regular definition, Transition diagrams, LEX.

UNIT-II

Syntax analysis: CFGs, Ambiguity, Associativity, Precedence, Top Down Parsing, Recursive Descent Parsing, Transformation on the grammars, Predictive Parsing, Bottom Up Parsing, Operator Precedence grammars, LR parsers (SLR, LALR, LR), YACC.

Syntax directed definitions: Inherited and Synthesized Attributes, Dependency Graph, Bottom Up and Top Down Evaluation of Attributes, L- and S-Attributed Definitions.

UNIT-III

Type checking: Type System, Type Expressions, Structural and Name Equivalence of types, Type Conversion, Overloaded Functions and Operators.

Run time system: Storage Organization, Activation Tree, Activation Record, Parameter Passing, Symbol Table, Dynamic Storage Allocation.

UNIT-IV

Intermediate code generation: Intermediate Representations, Translation of Declarations, Assignments, Control Flow, Boolean Expressions and Procedure Calls.

Code generation and instruction selection: Issues, Basic Blocks and Flow Graphs, Register Allocation, Code Generation, DAG representation of programs, Code Generation from DAGS, Peep-Hole Optimization, Code Generators, Specifications of machine.

Suggested Readings:

1. Aho, A.V., Sethi, R. and Ullman, J.D., *Compilers: Principles, Techniques and Tools*, Pearson, 2008.
2. Appel, A.W., *Modern Compiler Implementation in C*, Cambridge University Press, 2004.
3. Dhamdhere, D.M., *Compiler Construction – Principles & Practice*, Macmillan India, 2008.
4. Fischer, C.N. and LeBlanc Jr, R.J., *Crafting a Compiler*, Pearson, 2011.
5. Holub, A.I., *Compiler Design in C*, PHI, 1992.

Artificial Intelligence and Expert System
Course Code: SBS CS 01 03 19 C 4004

L T P **Total Credits: 4**
4 0 0

Course Objectives:

The primary objective of this course is to introduce the basic principles, techniques, and applications of Artificial Intelligence. Emphasis will be placed on the teaching of these fundamentals and labs for the 'hands-on' approach for understanding, as well as a challenging avenue for exploration and creativity.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Choose problems that are amenable to solution by AI methods, and which AI methods may be suited to solving a given problem.
- Examine a given problem in the language/framework of different AI methods
- Describe basic AI algorithms (e.g., standard search algorithms or resolution).
- Conclude and carry out an empirical evaluation of different algorithms on a problem formalization.

UNIT-I

Basic Concepts: AI and its importance, history of AI, applications areas, AI approach for solving problems. Problem representation: State space representation, problem reduction representation, bounding functions. Propositional logic: syntax and semantics. First order predicate logic (FOPL): syntax and semantics, conversion to clausal form, inference rules, unification, resolution principle, proof procedure, refutation.

UNIT-II

Search and Control Strategies: Strategies for state space search, data driven and goal driven search; Search algorithms- uninformed search (depth first, breadth first, depth first with iterative deepening) and informed search (Hill climbing, best first, A, A*, AO algorithm, mini-max etc.), computational complexity, Properties of search algorithms- Admissibility, Monotonicity, Optimality, Dominance, etc. , genetic algorithms.

UNIT-III

Expert System Architecture: Rule based architecture, Non-production system architecture. Components of Expert Systems, Stages of expert system development, Expert systems applications, Building Expert System and Shell. Knowledge acquisition and validation.

UNIT-IV

Managing uncertainty in expert systems: Bayesian probability theory, Stanford certainty factor algebra, No monotonic logic and reasoning with beliefs, Fuzzy logic, Dempster/Shaffer theory.

Suggested Readings:

1. Luger, G.F. and Stubblefield, W.A., *Artificial Intelligence and The Design of Expert Systems*, Benjamin-Cummings Publishing Co. Inc., 2008.
2. Nilsson, N.J., *Principles of Artificial Intelligence*, Morgan Kaufmann, 2014.
3. Patterson, D.W., *Introduction to Artificial Intelligence and Expert Systems*, Prentice-hall of India, 2007.
4. Rich, E.K. and Nair, S.B., *Artificial Intelligence*, New Delhi, 2009.
5. Russell, S., and Norvig, P., *Artificial Intelligence: A Modern Approach*, Prentice Hall, 2015.

Information and Network Security
Course Code: SBS CS 01 03 20 C 4004

L T P **Total Credits: 4**

4 0 0

Course Objectives:

The purpose of this course is to provide understanding of the main issues related to security in modern networked computer systems. This covers underlying concepts and foundations of computer security, basic knowledge about security-relevant decisions in designing IT infrastructures, techniques to secure complex systems and practical skills in managing a range of systems, from personal laptops to large-scale infrastructures.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Explore the concept of network security
- Introduce the concept of security in this information world.
- Identify malicious software and network threats.
- Apply different aspects of information security and its applications.

UNIT-I

Information Security Concepts- Introduction to Information Security, Meaning of security, attacks, Introduction to Information and Network Security, Integrity, and Availability, NIST FIPS 199 Standard, Assets and Threat Models, Examples

Least privilege, access control, operating system security- The principle of least privilege, Access control concepts, Operating system mechanisms, Unix, Windows, and Android examples.

UNIT-II

Cryptography: Algorithms for block and stream ciphers, private key encryption – DES, AES, RC4; Algorithms for public key encryption – RSA, DH Keyexchange, KERBEROS, elliptic curve cryptosystems.

UNIT-III

Network Security- Network concepts, Threats in Networks, Threats in Transit, TCP/IP security issues, Impersonation, DNS security issues and defenses, Network security tools, Secure protocols, Firewalls, VPNs, Tor, I2P, Intrusion Detection and filters, Host-Based IDS vs Network-Based IDS, Dealing with unwanted traffic: Denial of service attacks.

UNIT-IV

Malicious Software and Software Security- Malicious Web, Internet Security Issues, Types of Internet Security Issues, Viruses, Trojans, rootkits, worms, botnets, Spyware, Key-Loggers, Ransomware, Honeypot, Security policies, penetration testing, Sandboxing, buffer overflow vulnerability and attack, control hijacking attacks, mobile security, Repackaging attacks, Attacks on mobile apps, Whole-disk encryption.

Suggested Readings:

1. Goodrich, M.T. and Tamassia, R., *Introduction to Computer Security*, Pearson Publication, 2011.
2. Gupta, B., Agrawal, D.P. and Yamaguchi, S., *Handbook of Research on Modern Cryptographic Solutions for Computer and Cyber Security,* IGI Global, 2016.
3. Menezes, A.J., Katz, J., Oorschot, P.C.V. and Vanstone, S.A., *Handbook of Applied Cryptography*, CRC Press, 2018.
4. Perlman, R., Kaufman, C. and Speciner, M., *Network Security: Private Communication in a Public World*, Pearson Education, 2016.
5. Stallings, W., *Network Security Essentials: Applications and Standards*, Pearson Education India, 4th Edition, 2011.

Internet and Java Programming
Course Code: SBS CS 01 03 21 C 4004

L T P **Total Credits: 4**
4 0 0

Course Objectives:

The objective of this course is to teach students about programming in the Java language and the use of Java in a variety of technologies and on different platforms.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Describe the fundamental blocks of the Internet and TCP/IP.
- Identify classes, objects, members of a class and relationships among them needed for a specific problem
- Write Java application programs using OOP principles and proper program structuring
- Demonstrate the concepts of polymorphism, inheritance and error handling techniques using exception handling

UNIT-I

Internetworking with TCP / IP: Review of network technologies, Internet addressing, Address resolution protocols (ARP / RARP), Routing IP datagrams, Reliable stream transport service (TCP) TCP / IP over ATM networks, Internet applications - E-mail, Telnet, FTP, NFS, Internet traffic management.

UNIT-II

The overview of Java's architecture and the architecture of the Java Virtual Machine (JVM).
Methods: Method Declarations, this reference, Method Overloading, Constructors, The Default Constructor and Constructors overloading. Arrays, Anonymous Arrays, Multidimensional Arrays, The main() Method, Program Arguments.
Classes: Declaring Members (Fields and Methods), Instance Members, Static Members.
Objects: Class Instantiation, Reference Values, and References. Primitive Data Types, Variable Declarations, Initial Values for Variables, Class Declarations.

UNIT-III

Object-Oriented Programming: Single Implementation Inheritance, Overriding Methods, Hiding Members, The Object Reference super, Chaining Constructors Using this() and super().

Interfaces: Defining Interfaces, Abstract Method Declarations, Implementing Interfaces, Extending Interfaces, Interface References, Constants in Interfaces, Polymorphism and Dynamic Method Lookup.

UNIT-IV

Exception Handling: The try Block, the catch Block, the finally Block, the throw Statement, the throws Clause, Checked and Unchecked Exceptions, Defining New Exceptions.

Multithreading: Overview of Threads, the Main Thread, Thread Creation, Synchronization, Thread Transitions. Basics of Event Handling, Graphics Programming using AWT and Swing.

Suggested Readings:

1. Comer, D.E., Stevens, D.L. and Evangelista, M., *Internetworking with TCP/IP, Vol. III: Client-Server Programming and Applications*, Prentice Hall, 2001.
2. Deitel, P. and Deitel, H., *Java How to Program*, Pearson Education, 2015.
3. Eckel, B., *Thinking in Java*, Pearson Education, 2006.
4. Freeman, A. and Ince, D., *Programming the Internet with Java*, Addison-Wesley Longman Publishing Co. Inc., 1998.
5. Horstmann, C.S. and Cornell, G., *Core Java Volume I (Fundamentals)*, Pearson, 2019.

Network Programming
Course Code: SBS CS 01 03 09 E 3003

L T P **Total Credits: 3**
3 0 0

Course Objectives:

The objective of this course is to teach how to write network programs using an application program interface (or API), implement basics of socket programming using TCP Sockets. This course will guide the students to create client and server applications using the "Sockets" API and the implementation of Data link layer protocol and TCP layer.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Understand the key protocols that support the Internet.
- Have a detailed knowledge of the TCP/UDP Sockets.
- Learn advanced programming techniques such as Broadcasting, Multicasting.
- Analyse the security requirements of a networked programming environment and identify the issues to be solved.

UNIT-I

Introduction to Network Programming: OSI model, Unix standards, TCP and UDP & TCP connection establishment and Format, Buffer sizes and limitation, standard internet services, Protocol usage by common internet application.

Sockets: Address structures, value – result arguments, Byte ordering and manipulation function and related functions Elementary TCP sockets – Socket, connect, bind, listen, accept, fork and exec function, concurrent servers. Close function and related function.

UNIT-II

TCP client server: Introduction, TCP Echo server functions, Normal startup, terminate and signal handling server process termination, Crashing and Rebooting of server host shutdown of server host.

I/O Multiplexing and socket options: I/O Models, select function, Batch input, shutdown function, poll function, TCP Echo server, getsockopt and setsockopt functions. Socket states, Generic socket option IPV6 socket option ICMPV6 socket option IPV6 socket option and TCP socket options.

UNIT-III

Elementary UDP sockets: Introduction UDP Echo server function, lost datagram, summary of UDP example, Lack of flow control with UDP, determining outgoing interface with UDP.

Elementary name and Address conversions: DNS, gethost by Name function, Resolver option, Function and IPV6 support, uname function, other networking information.

UNIT-IV

IPC: Introduction, File and record locking, Pipes, FIFOs streams and messages, Name spaces, system IPC, Message queues, Semaphores.
Remote Login: Terminal line disciplines, Pseudo-Terminals, Terminal modes, Control Terminals, rlogin Overview, RPC Transparency Issues.

Suggested Readings:

1. Chan, T., *Unix System Programming using C++*, Prentice Hall, 1999 .
2. Glass, G. and Ables, K., *UNIX for Programmers and Users*, Prentice Hall, 2003.
3. Richard, S.W., *Unix Network Programming. In The Sockets Networking API (Vol. 1)*, Pearson Education India, 2015.
4. Rochkind, M.J., *Advanced UNIX Programming*, Pearson Education, 2004.
5. Stevens, W.R., Rudoff, A.M. and Fenner, B., *UNIX Network Programming Volume 1: The Sockets Networking API (Vol. 3)*, Addison-Wesley Professional, 2003.

Machine and Deep Learning
Course Code: SBS CS 01 03 10 E 3003

L T P **Total Credits: 3**
3 0 0

Course Objectives:

This course will serve as a comprehensive introduction to various topics in machine learning. At the end of the course, the students should be able to design and implement machine learning solutions to classification, regression, and clustering problems; and be able to evaluate and interpret the results of the algorithms.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Apply supervised and unsupervised learning algorithms.
- Describe the fundamental concepts in machine learning and popular machine learning algorithms.
- Solve problems related to the application of machine learning algorithms with programming in Python/ MatLab.
- Describe the concept of Deep Learning

UNIT-I

Introduction: History of Machine Learning, Programs vs learning algorithms, Machine Learning definition, types of learning, hypothesis space and inductive bias, evaluation, cross-validation, Instance based learning, Feature reduction, Collaborative filtering-based recommendation, Gradient Descent learning.

UNIT-II

Supervised Learning: General notions - Bayes optimality, curse of dimensionality, overfitting and model, selection, bias vs. variance tradeoff, generative vs. discriminative for parameter estimation, feature selection, and etc Linear methods - linear, logistic regression and generalized linear models, naive Bayes, linear discriminant analysis, support vector machines

Nonlinear methods - kernel methods, nearest neighbor, decision trees, neural networks, and etc Ensemble learning - bagging, boosting, and etc.

UNIT-III

Unsupervised Learning: Clustering and density estimations - K-means/vector quantization, mixture models, Dimensionality reduction - linear and nonlinear methods, Principal components analysis.

Deductive Learning: Probability theory and Bayes rule. Naive Bayes learning algorithm. Parameter smoothing. Generative vs. discriminative training. Logistic regression. Bayes nets and Markov nets for representing dependencies.

UNIT-IV

Deep Learning: Artificial Neural Networks, Perceptron, Multilayer networks and Backpropagation algorithm, Introduction to Deep Neural networks, Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs).

Suggested Readings:

1. Alpaydin, E., *Introduction to Machine Learning*, MIT Press, 2004.
2. Bishop, C.M., *Pattern Recognition and Machine Learning,* Springer, 2006.
3. Hastie, T., Tibshirani, R. and Friedman, J., *The Elements of Statistical Learning*, Springer, 2008.
4. Mitchell, T.M., *Machine Learning,* McGraw- Hill, 1997.
5. Russell, S. and Norvig, P., *Artificial Intelligence: A Modern Approach,* Prentice Hall, 2002.

Software Project Management
Course Code: SBS CS 01 03 11 E 3003

L T P **Total Credits: 3**
3 0 0

Course Objectives:

This course is aimed at introducing the important concepts of project management related to managing software development. Students will also get familiar with the different activities involved in Software Project Management. Further, they will also come to know how to successfully plan and implement a software project management activity, and to complete a specific project in time with the available budget.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Identify the need for Software Project Management.
- Identify different techniques for software cost estimation
- Understand activity planning and risk management
- Manage and control projects.

UNIT-I

Project Evaluation and Project Planning: Importance of Software Project Management, Activities Methodologies, Categorization of Software Projects, Setting objectives, Management Principles, Management Control, Project portfolio Management, Cost-benefit evaluation technology, Risk evaluation, Strategic program Management, Stepwise Project Planning.

UNIT-II

Project Life Cycle and Effort Estimation: Software process and Process Models, Choice of Process models, Mental delivery, Rapid Application development, Agile methods, Extreme Programming, SCRUM, Managing interactive processes, Basics of Software estimation, Effort and Cost estimation techniques, COSMIC Full function points, COCOMO II- A Parametric Productivity Model, Staffing Pattern.

UNIT-III

Activity Planning and Risk Management: Objectives of Activity planning, Project schedules, Activities, Sequencing and scheduling, Network Planning models, Forward Pass & Backward Pass techniques, Critical path (CRM) method, Risk identification, Assessment, Monitoring,

PERT technique, Monte Carlo simulation, Resource Allocation, Creation of critical patterns, Cost schedules.

UNIT-IV

Project Management and Control: Framework for Management and control, Collection of data Project termination, Visualizing progress, Cost monitoring, Earned Value Analysis, Project tracking, Change control, Software Configuration Management, Managing contracts, Contract Management.

Suggested Readings:

1. Futrell, R.T., Shafer, L.I. and Shafer, D.F., *Quality Software Project Management*, Prentice Hall, 2001.
2. Meredith, J.R., Shafer, S.M. and Mantel S.J., *Project Management: A Strategic Managerial Approach*, John Wiley & Sons, 2017.
3. Royce, W., *Software Project Management*, Pearson Education India, 1998.
4. Stellman, A. and Greene, J., *Applied Software Project Management*, O'Reilly Media, Inc., 2005.
5. Wysocki, R.K., *Effective Software Project Management*, John Wiley & Sons, 2010.

Digital Image Processing
Course Code: SBS CS 01 03 12 E 3003

L T P **Total Credits: 3**
3 0 0

Course Objectives:

In this course, students will understand the image fundamentals and mathematical transforms necessary for image processing. Study about the image enhancement techniques, image restoration procedures and image compression procedures. Students will learn about the image segmentation and representation techniques & pattern recognition and interpretation.

Course Level Learning Outcomes

On completion of the module the student will be able to:

- Explain and analyze the steps of image formation, sampling, quantization and representation digitally.
- Outline and examine how images are processed by discrete, linear, time-invariant systems.
- Summarize how images are perceived by humans and how colour is represented.
- Point out how image information can be modeled analytically and compare transform-domain representation of images (Fourier, DCT, Haar, WHT).

UNIT-I

Digital Image Fundamentals: Digital Image Processing, Origins of Digital Image Processing Application of Digital Image Processing, Steps in Digital Image Processing, Components of an Image Processing System, Image formation, Image transforms – Fourier transforms

UNIT-II

Image Enhancement Techniques: Histogram modification techniques - Image smoothening Image Sharpening - Image Restoration - Degradation Model – Noise models - Spatial filtering – Frequency domain filtering.

UNIT-III

Image Compression & Segmentation: Compression Models - Elements of information theory Error free Compression -Image segmentation –Detection of discontinuities , Region based segmentation - Morphology.

UNIT-IV

Representation and Description: Representation schemes- Boundary descriptors- Regional descriptors - Relational Descriptors.

Suggested Readings:

1. Christopher, R.M., *An Introduction to MATLAB for Behavioral Researchers.*, SAGE Publications, Inc., 2013
2. Forsyth, D., and Ponce, J., *Computer Vision: A Modern Approach.*, Pearson Education India., 2015.
3. Gonzalez, C., Rafael., E., and Woods, R., *Digital Image Processing*., Pearson Publication., 2007.
4. Gonzalez, R., Woods, R., and Eddins, S., *Digital Image Processing Using MATLAB*., McGraw Hill Education., 2017.
5. Jayaraman, S., Veerakumar, T., and Esakkirajan, S., *Digital Image Processing*., McGraw Hill Education., 2017.

Data Warehousing and Data Mining
Course Code: SBS CS 01 03 13 E 3003

L T P **Total Credits: 3**
3 0 0

Course Objectives:

The main objective of this course is to impart the knowledge on how to implement classical models and algorithms in data warehousing and data mining and to characterize the kinds of patterns that can be discovered by association rule mining, classification and clustering. Data quality and methods and techniques for preprocessing of data.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Comprehend the various architectures and its application with data mining
- Design and develop data mining algorithms to analyze raw real world data
- Monitor and analyze to predict online digital activities
- Evaluate various mining techniques on complex data objects

UNIT-I

Basic Concepts, Data Warehousing Components, Building a Data Warehouse, Database Architectures for Parallel Processing, Parallel DBMS Vendors, Multidimensional Data Model, Data Warehouse Schemas for Decision Support, Concept Hierarchies, Characteristics of OLAP Systems, Typical OLAP Operations, OLAP and OLTP.

UNIT-II

Introduction to Data Mining Systems, Knowledge Discovery Process, Data Mining Techniques, Issues, Applications, Data Objects and attribute types, Statistical description of data, Data Preprocessing, Cleaning, Integration, Reduction, Transformation and Discretization, Data Visualization, Data similarity and dissimilarity measures.

UNIT-III

Mining Frequent Patterns, Associations and Correlations, Mining Methods, Pattern Evaluation Method, Pattern Mining in Multilevel, Multi-Dimensional Space, Constraint Based Frequent Pattern Mining, Classification using Frequent Patterns

UNIT-IV

Decision Tree Induction, Bayesian Classification, Rule Based Classification, Classification by Back Propagation, Support Vector Machines, Lazy Learners, Model Evaluation and Selection, Techniques to improve Classification Accuracy, Clustering Techniques, Cluster analysis, Partitioning Methods, Hierarchical Methods, Density Based Methods.

Suggested Readings:

1. Berson, A. and Smith, S.J., *Data Warehousing, Data Mining & OLAP*, Tata McGraw Hill, 2017.
2. Han, J., Pei, J. and Kamber, M., *Data Mining Concepts and Techniques*, Elsevier, 2011.
3. Pujari, A.K., *Data Mining Techniques*, Universities Press, 2010.
4. Soman, K.P., Diwakar, S. and Ajay, V., *Insight into Data Mining Theory and Practice,* PHI, 2009.
5. Witten, I.H. and Frank, E., *Data Mining: Practical Machine Learning Tools and Techniques*, Elsevier, 2016.

Data Science with R Programming
Course Code: SBS CS 01 04 26 C 4004

L T P **Total Credits: 4**
4 0 0

Course Objectives:

In this course, students will have knowledge on accessing, storing and manipulating the huge data from different resources. Students will understand the working environment of Pig and Hive for processing the structured and unstructured data. & differentiate the RDBMS and Hive architectures, and implement queries to process the data.

Course Level Learning Outcomes:

Upon completion of the module the student will be able to:

- Learn the main R data structures – vector and data frame and import external data into R for data processing and statistical analysis.
- Import, review, manipulate and summarize data-sets in R.
- Explore data-sets to create testable hypotheses and identify appropriate statistical tests.
- Apply R programming from a statistical perspective.

UNIT-I

Introduction of Data Science: Introduction to data science, Data collection, integration, management, modeling, analysis, visualization, prediction and informed decision making, Big data definition, structured and unstructured data. Exploratory data analysis Components of Hadoop Eco System- Data Access and storage, Data Intelligence, Data Integration, Data Serialization, Monitoring, Indexing

UNIT-II

R Programming: Basic commands, graphics, indexing data, loading data, Data types in R: Numeric/character/logical; real/integer/complex, creation of new variables, vectors, matrices, data frames, and lists, accessing elements of a vector or matrix.

UNIT-III

Operations with R: import and export of files, for loop, repeat loop, while loop, if command, if else command. Graphics in R: the plot command, histogram, bar-plot, box-plot, points, lines, segments, arrows, inserting mathematical symbols in a plot, pie diagram matrix operations such as addition, subtraction, multiplication, rank, eigenvalues, matrix inverse, generalized inverse, solution of linear equations.

UNIT-IV

Statistics Techniques and R: measures of central tendency and dispersion. Covariance, correlation, regression, some discrete and continuous probability z and t tests, F test for equality of variances, Chi-square tests.

Suggested Readings:

1. Braun, W.J. and Murdoch, D.J., *A First Course in Statistical Programming with R.*, Cambridge University Press., 2008.
2. Hadley, W., *Advanced R.*, Chapman and Hall/CRC Press, 2019.
3. Jones, O., Maillardet, R. and Robinson, A., *Introduction to Scientific Programming and Simulation Using R.*, Chapman and Hall/CRC., 2014.
4. Rhys, H.I., *Machine Learning with R, The Tidyverse, and MLR.*, Manning Publications., 2020.
5. Zumel, N., Mount, J. and Porzak, J., *Practical Data Science with R.*, Manning Publications., 2019.

Wireless Sensor Network and Internet of Things
Course Code: SBS CS 01 04 27 C 4004

L T P **Total Credits: 4**
4 0 0

Course Objectives:

The objective of this course is to impart comprehensive knowledge of the wireless networks needed for design and implementation of a typical IoT system. The course also aims at developing necessary skills required for efficient network infrastructure of an IoT. This course focuses on the latest microcontrollers with application development, product design and prototyping. This also focuses on interoperability in IoT along with various IoT Platforms for application development.

Course Level Learning Outcomes:

Upon completion of the module the student will be able to:

- Understand various concepts of ubiquitous sensing
- Apply Wireless Sensor Networks Principles in IoT
- Understand the role of Big Data, Cloud Computing and Data Analytics in a typical IoT system.
- Design a simple IoT system made up of sensors, wireless network connection, data analytics and display/actuators, and write the necessary control software.

UNIT-I

Introduction to Wireless Sensor Network: Wireless Sensor, Coverage & Placement, Topology Management in Wireless Sensor Network, Mobile WSNs, Medium Access Control in Wireless Networks, Routing in WSNs, Enabling Technologies for WSNs

UNIT-II

Architecture of Wireless Sensor Network: Sensor Network Scenarios, Optimization Goals, Figures of Merit, Design Principles for WSNs, Service Interfaces of WSNs Gateway Concepts Hardware Components, Energy Consumption of Sensor Nodes, Operating Systems and Execution Environments

UNIT-III

Internet of Things(IOT): Sensing, Actuation, Basics of Networking, Communication Protocols Sensor Networks, Machine to Machine Communications. Understanding of the IoT ecosystem, various layers in building an IoT application and interdependencies.

UNIT-IV

Applications of IOT & Arduino : IoT Introduction to Arduino Programming, Integration of Sensors and Actuators with Arduino, IoT platforms like PTC Thingworx and IoT

frameworks like MS Azure, usage of these platforms to build applications like Smart Cities and Smart Homes, Connected Vehicles, Smart Grid, Case Study: Agriculture, Healthcare

Suggested Readings:

1. Hersent, O., Boswarthick, D. and Elloumi, O., *Internet Of Things: Key Applications and Protocols.*, Wiley Publication., 2015.
2. Holler, J., Tsiatsis, V., Mulligan, C., Avesand, S., Karnouskos, S. and Boyle, D., *From Machine to Machine to the Internet of Things: Introduction to a New Age of Intelligence*., Academic Press., 2014.
3. Raj, P. and Raman, A.C., *The Internet of Things: Enabling Technologies, Platforms, and Use Cases.*, Auerbach Publications., 2017.
4. Yasuura, H., Kyung, C.M., Liu, Y. and Lin, Y.L., *Smart Sensors at the IoT Frontier.*, Springer Publication., 2018.
5. Zheng, J. and Jamalipour, A., *Wireless Sensor Networks: A Networking Perspective.*, Wiley Publication., 2014.

Distributed and Cloud Computing

Course Code: SBS CS 01 04 14 E 3104

L T P **Total Credits: 4**
3 1 0

Course Objectives:

Cloud computing is a key distributed systems paradigm that has grown popular in the last few years. Cloud technologies are pervasive, touching our daily lives any time we access the world wide web, use a mobile app, or make a retail purchase. The objective of this course is to teach the fundamental concepts of how and why Cloud systems works, as well as Cloud technologies that manifest these concepts such as Amazon AWS, Microsoft Azure and OpenStack etc.. Students will learn distributed systems concepts like virtualization, data parallelism.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Describe system models for distributed and cloud computing.
- Describe and distinguish different virtualization techniques.
- Explain cloud-enabling technologies, cloud mechanisms, and cloud architectures.
- Demonstrate the usage cloud application like AWS.

UNIT-I

Introduction to Distributed System: Models and Enabling Technologies, Computer Clusters for Scalable Parallel Computing, Virtual Machines and Virtualization of Clusters and Data centers.

UNIT-II

Introduction to Cloud Computing: Cloud Computing in a Nutshell System Model for Distributed and Cloud Computing, Roots of Cloud Computing, Grid and Cloud, Layers and Types of Clouds, Desired Features of a Cloud, Basic Principles, of Cloud Computing, Challenges and Risks, Service Models

UNIT-III

Virtual Machines and Virtualization of Cluster and Data Centres: Levels of Virtualization, Virtualization structures/Tools and Mechanism, Virtualization of CPU, Memory and I/O Devices, Virtual Clusters and Resources Management, Virtualization Data-Centre Automation

Service Oriented Architecture for Distributed Computing: Services & SOA, Message Oriented Middleware, Workflow in SOA.

UNIT-IV

Cloud Security, Data Security in the Cloud: An Introduction to the Idea of Data Security, The Current State of Data Security in the Cloud CryptDb: Onion Encryption layers- DET, RND, OPE, JOIN, SEARCH, HOM, and Homomorphism Encryption, FPE.

Suggested Readings:

1. Coulouris, G.F., Dollimore, J. and Kindberg, T., *Distributed Systems: Concepts and Design*, Pearson education, 2005.
2. Erl, T., Puttini, R. and Mahmood, Z., *Cloud Computing: Concepts, Technology, & Architecture*, Pearson Education, 2013.
3. Hwang, K., Dongarra, J. and Fox, G.C., *Distributed and Cloud Computing: From Parallel Processing to The Internet of Things*, Morgan Kaufmann, 2013.
4. Tanenbaum, A.S. and Van Steen, M., *Distributed systems: principles and paradigms*, Prentice-Hall, 2007.
5. Velte, T., Velte, A. and Elsenpeter, R., *Cloud Computing, A Practical Approach*, McGraw-Hill, Inc., 2009.

Bioinformatics
Course Code: SBS CS 01 04 15 E 3104

L T P **Total Credits: 4**
3 1 0

Course Objectives:

The unprecedented increase in the amount of available biological data ranging from protein sequences to biomedical images have rendered the use of computers and computational techniques for analysing and managing the biological data inevitable. This course aims to provide students with the basics of bioinformatics algorithms that have been applied over various types of biological data.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Describe about the different types of Biological databases.
- Describe about pairwise sequence alignment , algorithms and tools for pairwise alignment
- Describe about protein folding and its significance
- Apply algorithms and analysis methods to the real-world problems.

UNIT-I

Introduction to Bioinformatics: What is a Database, Types of Databases, Biological Databases, Pitfalls of Biological Databases, Information Retrieval from Biological Databases.

UNIT-II

Sequences: Problem statement, Edit distance and substitution matrices, HMMs and pairwise HMMs, Global and local alignments, Spliced alignment, Space-efficient sequence alignment, Multiple alignment, Database searching tools, Sequence by hybridization, Profile HMMs.

UNIT-III

Structures: Protein structure alignment, Protein Structure Prediction: Methods for predicting the secondary and tertiary structure of proteins. Techniques: neural networks, SVMs, genetic algorithms and stochastic global optimization.

UNIT-IV

Transcriptomics: Methods for analysing gene expression and microarray data. Techniques: clustering, SVMs. Agent-based Genome Analysis: Automation of genome analysis using intelligent software agents. Drug Discovery Informatics: Approaches to drug discovery using bioinformatics techniques.

Suggested Readings:

1. Compeau, P. and Pevzner, P.A., *Bioinformatics Algorithms: An Active Learning Approach*, Active Learning Publishers, 2018.
2. Jones, N.C., Pevzner, P.A. and Pevzner, P., *An Introduction to Bioinformatics Algorithms*, MIT press, 2004.
3. Krawetz, S.A. and Womble, D.D., *Introduction to Bioinformatics: A Theoretical and Practical Approach*, Springer Science & Business Media, 2003.
4. Lesk, A., *Introduction to bioinformatics*, Oxford University Press, 2019.
5. Mandoiu, I. and Zelikovsky, A., *Bioinformatics Algorithms: Techniques and Applications*, John Wiley & Sons, 2008.

Natural Language Processing and Speech Recognition

Course Code: SBS CS 01 04 16 E 3104

L T P **Total Credits: 4**

3 1 0

Course Objectives:

The objective of this course is to give students a clear understanding of linguistics methods, various tools and aspects of NLP like syntax and semantic analysis, parsing, machine translation, information retrieval and statistical discourse processing.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Describe the challenges involved in developing NLP solutions.
- Describe various recent statistical methods in natural language processing.
- Develop the linguistics and their application to part-of-speech tagging.
- Develop background to various tools and aspects of NLP like syntax and semantic analysis, parsing, machine translation, information retrieval and statistical discourse processing

UNIT-I

Introduction: NLP tasks in syntax, semantics, and pragmatics. Applications such as information extraction, question answering, and machine translation. The problem of ambiguity. The role of machine learning. Brief history of the field.

UNIT-II

N-gram Language Models: The role of language models. Simple N-gram models. Estimating parameters and smoothing. Evaluating language models.

Part of Speech Tagging and Sequence Labeling: Lexical syntax. Hidden Markov Models (Forward and Viterbi algorithms and EM training)

UNIT-III

Syntactic parsing: Grammar formalisms and treebanks. Efficient parsing for context-free grammars (CFGs). Statistical parsing and probabilistic CFGs (PCFGs). Lexicalized PCFGs. Neural shift-reduce dependency parsing

Semantic Analysis: Lexical semantics and word-sense disambiguation. Compositional semantics. Semantic Role Labeling and Semantic Parsing.

UNIT-IV

Mechanics of Speech: Speech Production Mechanism, Nature of Speech Signal, Discrete Time Modeling of Speech Production, Representation of Speech Signals, Classification of Speech Sounds, Phones, Phonemes, Phonetics, IPA and Phonetic Alphabets, Articulatory Features, Auditory Perceptions, Anatomical Pathways from Ear to the Perception of Sound Peripheral Auditory System.

Suggested Readings:

1. Jurafsky, D. and Martin, J.H., *Speech and Language Processing*, Pearson, 2020.
2. Lane, H., Howard, C. and Hapke, H., *Natural Language Processing in Action*, Manning Publications, 2019.
3. Manning, C. and Schutze, H., *Foundations of Statistical Natural Language Processing*, MIT Press, 1999.
4. Rabiner, L.R. and Juang, B.H., *Fundamentals of Speech Recognition*, Pearson, 2009
5. Thanaki, J., *Python Natural Language Processing*, Packt, 2017.

Computer Vision
Course Code: SBS CS 01 04 17 E 3104

L T P **Total Credits: 4**
3 1 0

Course Objectives:

The objectives of this course are to develop the understanding of the basic principles and techniques of image processing and image understanding, and to develop your skills in the design and implementation of computer vision software. This course will explore some of the basic principles and techniques from these areas which are currently being used in real-world computer vision systems and the research and development of new systems.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Describe image representation and basics of computer vision.
- Implement fundamental image processing for vision application.
- Implement the pattern recognition methods.
- Develop various applications using computer vision.

UNIT-I

Digital image representation, image acquisition, storage and processing. fundamental steps in image processing, Introduction to Image Processing, Computer Vision and Computer Graphics, Monocular imaging system, basics of Image Formation, Radiance, Irradiance, BRDF, color etc. Binocular imaging systems, Multiple views geometry, Structure determination, shape from shading and Photometric Stereo.

UNIT-II

Introduction, definition, Active vision system, Machine vision components, hardware's and algorithms, segmentation, data reduction, feature extraction, edge detection, image recognition and decisions, identification, Triangulation geometry, resolution passive and active stereo imaging, optical scanners, interfacing machine vision system, vision system calibration.

UNIT-III

Structure of the human eye, image formation, brightness adaptation and discrimination, a simple image model, uniform and non-uniform sampling and quantization, distance measures, Fourier and wavelet descriptors, Multiresolution analysis, Hough transforms and other simple object recognition methods, PCA, HMM and GMM.

UNIT-IV

Face detection and Face recognition, Eigen faces, Active appearance and 3D shape models of faces. Surveillance – foreground-background separation, particle filters, Chamfer matching, tracking, and occlusion, combining views from multiple cameras, human gait analysis.

Suggested Readings:

1. Davies, E. R., *Computer & Machine Vision,* 4th Edition, Academic Press, 2012.
2. Forsyth, D.A. and Ponce, J., *Computer Vision: A Modern Approach*. Prentice Hall Professional Technical Reference, 2002.
3. Mark, N.A.S., *Feature Extraction & Image Processing for Computer Vision*, 3rd Edition, Academic Press, 2012.
4. Prince, S.J., *Computer Vision: Models, Learning, and Inference*, Cambridge University Press, 2012.
5. Szeliski, R., *Computer Vision: Algorithms and Applications (CVAA)*. Springer Science and Business Media, 2010.

Embedded Programming

Course Code: SBS CS 01 04 18 E 3104

L T P **Total Credits: 4**
3 1 0

Course Objectives:

In this course, Student will understand the architecture of embedded processors, microcontrollers, and peripheral devices and appreciate the nuances of programming micro-controllers in assembly for embedded systems. The challenges in developing operating systems for embedded systems. Students will learn about programming these systems in high-level languages such as C.

Course Level Learning Outcomes:

Upon completion of the module the student will be able to:

- Describe the differences between the general computing system and the embedded system, also recognize the classification of embedded systems.
- Explain the architecture of the ATOM processor and its programming aspects (assembly Level)
- Describe the interrupts, hyper threading and software optimization.
- Design real time embedded systems using the concepts of RTOS.

UNIT-I

Introduction of Embedded Systems: Concept of Embedded System Design: Design challenge, Processor technology, IC technology, Embedded Design technology.

UNIT-II

Single and General Purpose Processor : introduction, basic architecture, operation, super-scalar and VLSIIW architecture, application specific instruction set processors (ASIPS), microcontrollers, digital signal processors, selecting a microprocessor.

UNIT-III

Memory and Input / Output Management: Interfacing Analog and digital blocks: Analog-to-Digital Converters (ADCs), Digital to-Analog, Converters (DACs)., Communication basics and basic protocol concepts, Microprocessor interfacing: I/O addressing, Port and Bus based, I/O, Memory mapped I/O, Standard I/O interrupts, Direct memory access , communication principles parallel, serial and wireless.

UNIT-IV

Processes and Operating Systems: Real time operating systems, Kernel architecture: Hardware, Device Embedded operating systems, Task scheduling in embedded systems: task scheduler, first in first out, shortest job first, round robin, priority based scheduling. Types of embedded operating systems.

Suggested Readings:

1. Barrett, S.F. and Pack, D.J., *Embedded Systems: Design and Applications.*, Pearson Education India, 2008.
2. Barry, P. and Crowley, P., *Modern Embedded Computing: Designing Connected, Pervasive, Media-Rich Systems*, Morgan Kaufmann Publication, 2012.
3. Kamal, R., *Embedded Systems: Architecture, Programming and Design.*, 2nd Edition., McGraw Hill Education, 2008.
4. Shibu, K.V., *Introduction to Embedded Systems.*, McGraw Hill Education, 2017.
5. Vahid, F. and Givargis, T., *Embedded System Design: A Unified Hardware / Software Introduction.*, Wiley Publication, 2006.

Web Development using PHP Lab-III
Course Code: SBS CS 01 01 08 C 0021

L T P **Total Credit: 1**
0 0 2

Course Objectives:

The objective of this course is to provide the necessary knowledge to design and develop dynamic, database-driven web applications using PHP version 5. Students will learn how to connect to any ODBC-compliant database, and perform hands on practice with a MySQL database to create database-driven HTML forms and reports etc. Students also learn how to configure PHP and Apache Web Server. Comprehensive lab exercises provide facilitated hands on practice crucial to develop competence web sites.

Course Level Learning Outcomes:

Upon successful completion of the course students will be able to:

- Build a simple, yet functional web application using PHP/MySQL.
- Setup and configure MySQL, PHP, and web server development environment.
- Select, insert, update and delete data using SQL language.
- Use PHP built-in functions and creating custom functions.

Android Application Development Lab-III
Course Code: SBS CS 01 03 24 C 0021

L T P **Total Credit: 1**
0 0 2

Course Objectives:
Android Application Development course is designed so that students are able to write apps for Android devices. The student will learn the basics of Android platform and get to understand the application lifecycle. Covers Android application development phases, terminologies, application design, and coding. Students will be able to write simple GUI applications, use built-in widgets and components, work with the database to store data locally, and much more.

Course Level Learning Outcomes:
Upon successful completion of the course students will be able to:

- Install and configure Android application development tools.
- Design and develop user Interfaces for the Android platform.
- Save state information across important operating system events.
- Apply Java programming concepts to Android application development.

Advance Java Programming Lab-II
Course Code: SBS CS 01 04 29 C 0021

L T P **Total Credit: 1**
0 0 2

Course Objectives:
In this course, student will learn using Graphics, Animations and Multithreading for designing Simulation and Game based applications. Student will get in touch with design and develop GUI applications using Abstract Windowing Toolkit (AWT), Swing and Event Handling. Design and develop Web applications. Student will understand about designing Enterprise based applications by encapsulating an application's business logic. Designing applications using pre-built frameworks.

Course Level Learning Outcomes:
Upon successful completion of the course students will be able to:

- Learn to access database through Java programs, using Java Data Base Connectivity (JDBC)
- Create a full set of UI widgets and other components, including windows, menus, buttons, checkboxes, text fields, scrollbars and scrolling lists, using Abstract Windowing Toolkit (AWT) & Swings
- Use Internet Programming using Java Applets
- Map Java classes and object associations to relational database tables with Hibernate mapping files

(General Elective Paper)
Emerging Trends and Technologies in IT
Course code: SBS CS 01 02 19 E 3104

L T P **Total Credits: 4**
3 1 0

Course Objectives:
In this course, students enables to compare and contrast multiple division techniques, mobile communication systems, and existing wireless networks. Provide students with the fundamentals and essentials of Cloud Computing. Student will also get knowledge of protocols used in wireless communications and perform simulations of wireless networking.

Course Level Learning Outcomes:
Upon successful completion of the course students will be able to:

- Describe the possible future of mobile computing technologies and applications.
- Demonstrate basic skills for cellular networks design.
- Identify problems, and explain, analyse, and evaluate various cloud computing solutions.
- Compare different solutions for communications at each network layer

UNIT-I

Mobile Computing and the "Post-PC" Era: Smartphones & Tablets: Why now? MobilePlatforms (e.g. iOS, Android, BB, Windows), Applications Markets.

UNIT-II

Cloud Computing: What does ―X as a Service‖ mean (X=Platform, Infrastructure orSoftware), Service Models, Scalability and Reliability, Development frameworks (e.g. AWS, Azure App Engine) Business Benefits, Cloud Security & Privacy, Regulation, Consumerization of IT.

UNIT-III

Mobile Computing (MC): Introduction to MC, novel applications, limitations, andarchitecture. GSM: Mobile services, System architecture, Radio interface, Protocols, Localization and calling, Handover, Security, and New data services.

UNIT-IV

Medium Access Control (Wireless): Motivation for a specialized MAC (Hidden andexposed terminals, Near and far terminals), SDMA, FDMA, TDMA, CDMA. Mobile Network Layer, Mobile Transport LayerAudio and Video Conferencing: Technology & Applications, Application to informationtechnology to various function areas such as education, banking, communication etc.

Suggested Readings:

1. Chowdhury, M., Biswas, A.., *Wireless Communication: Theory and Applications*., Cambridge University Press., 2016.
2. Erl, T., Puttini, R., Mahmood, Z., *Cloud Computing: Concepts, Technology & Architecture*., Pearson Education India., 2014.
3. Jackson, W*., Android Studio New Media Fundamentals: Content Production of Digital Audio/Video, Illustration and 3D Animation*., Apress Publisher., 2016.
4. Talukder, K.A., Ahmed, H., Yavagal, R., *Mobile Computing: Technology, Applications and Service Creation*., 2nd edition., McGraw Hill., 2014.
5. Schiller, J., *Mobile Communications*., 2nd Edition., Pearson., 2014.