SYLLABUS

M.SC. (COMPUTER SCIENCE / INFORMATION TECHNOLOGY)

1st SEMESTER

Session 2020 - 2021

Mission of SCS&IT, DAVV

To produce world-class professionals who have excellent analytical skills, communication skills, team building spirit and ability to work in cross cultural environment.

To produce international quality IT professionals, who can independently design, develop and implement computer applications.

Professionals who dedicate themselves to mankind, who are environment conscious, follow social norms and ethics.

School of Computer Science & IT,
Devi Ahilya Vishwa Vidyalaya, Indore
www.scs.dauniv.ac.in

Course Name MSc (CS/IT) 1st Semester
Subject Code: IC-4916
Subject Name: Communication Skills and Report Writing

Aim of the Subject

To improve the confidence, communication skills and presentation capabilities of students that will help them in placements and corporate life.

Objectives

To develop effective communication skills in students which will help them in facing interviews and group discussions

Learning Outcomes

1. Improved skills in personal interviews and group discussions
2. Development of power of expression

Unit 1

Basics of Communication

Unit 2

Development of Group Discussion Skills

Unit 3

Development of Presentation Skills and facing interviews

Unit 4

Basics of Written Communication

Unit 5

Intense practice of Presentations, Group Discussions and Interviews

Text Book(s)

1. Communication – K. K. Sinha
2. Organizational Behavior - Fred Luthans
3. Organizational Behavior - Stephen Robbins

Reference Material(s)

1. Communications Skills – M.V. Rodrigues
2. Times of India/ Hindustan Times/ The Hindu etc.

Course Name MSc (CS/IT) 1st Semester
Subject Code: CS-5511
Subject Name: Operating Systems

Aim of the Subject

The course aims to explore the importance of the operating system and its function. The different techniques used by the operating system to achieve its goals as resource manager.

Objectives

1. To learn and understand the Concepts of operating system
2. To Learn and understand operating system services
3. The core structure, functions and design principles of operating system
4. Interposes communications and basic concepts of virtualization

Learning Outcomes

Students will be having understanding of following concepts of Operating System:

1. Process Management
2. Memory Management
3. File & I/O Management

Unit 1

Introduction: Evolution of operating systems, operating system concepts; activities, functions and services of operating system; Computer Systems: Mainframe, Desktop, Multiprocessors, Distributed, Clustered, Real time and Hand held systems. Computer System Operations, Storage hierarchy, Hardware protection, System calls, System structures. Process Management: Process concepts, Process scheduling, Operation on processes.

Unit 2

Cooperating processes, Inter-process communication. Threads: multithreading models, threading issues, thread examples. CPU Scheduling: concepts, scheduling criteria, scheduling algorithms, algorithm evaluation. Process synchronization: Critical section problem, Mutual exclusion and synchronization Techniques of inter process: Synchronization hardware, semaphore, classical problems of synchronization, critical regions and monitors. Deadlock: deadlock characterization, deadlock handling methods.

Unit 3

Memory Management: Concepts, single user memory management. Partition memory allocation: paging, segmentation and segmentation with paging, Virtual memory management: concept, demand paging, process creation, page replacement, allocation of frames and thrashing.

Unit 4

File Management: File concepts, access methods, directory structure, file system mounting, sharing and protection of files. File system structure and implementation, allocation methods, free space management, reliability of file system. Distributed file system and structures.

Unit 5

Device Management: Goals of input/output software design, Structure of device hardware and software. Layers of I/O software, structure of device drivers, Disk driver, disk arm scheduling algorithms, terminal driver, clock driver, Case study of Windows 2000.

Text Book(s)

A. Silberschatz, P. Galvin and G. Gagne, Operating System Concepts, Addison Wesley, 8th Edition, 2008.

Reference Material(s)

William Stallings, Operating Systems: Internals and Design Principles 4th Edition, Pearson Education, 2003.

Course Name MSc (CS/IT) 1st Semester
Subject Code: CS-4116
Subject Name: Discrete structure

Aim of the Subject

To give better understanding about the subject so that student are good in problen solving skills as well as they can understand few more subject that are based on Discrete structure.

Objectives

The objective of this course is to teach students how to think logically and mathematically.

Learning Outcomes

Students will learn the basic concepts of sets, permutations, relations, graphs, trees and finite state machines. Students will represent discrete objects and relationships using abstract mathematical structures.

Unit 1

The Foundations: Logic, Sets and Functions: Introduction to set theory, set operations, fuzzy
sets, mathematical logic, prepositions, prepositional equivalences, predicates and quantifiers.
Importance of Quantifiers. Functions, functions for computer science.
Mathematical reasoning: Introduction to Methods of proof, mathematical induction. Use of
mathematical induction to solve different problems. Importance of recursions in computer
science, scope of recursions, Recursive definitions, recursive algorithms.

Unit 2

Combinatorics: The basics of counting, The sum rule, The product rule, The Pigeonhole
Principle, Permutations with repetitions, Permutations without repetitions, Circular Permutations.
Applications of combinations. Applications of Combinatorics to solve Committee problems,
word problems, puzzle problems etc. Applications of Combinatorics to understand Telephone
numbering plan, understanding Internet addresses, Advanced counting techniques, recurrence
relations, solving recurrence relations, algorithm design, Basic understanding of complexities,
basic problems of complexity of algorithms.

Unit 3

Relations: Relation definition , Importance of relations in computer science, Relations and their properties, Unary relations , Binary relations, Ternary relations , n-ary relations and their applications, closures of relations, equivalence relations, partial ordering. Representing relations, relation matrix, relation graph, composite relation. Operations on relations – union, intersection and join. Concepts of least upper bond, Greatest lower bond, maximal element, minimal element, Greatest element, Least element of a partially ordered set, lattices, sub lattices, chains and antichains.

Unit 4

Graphs: Introduction to Graphs, Importance of graph theory in computer science, Graph terminology, representing graphs, graph types, graph models, and graph isomorphism. Connectivity, Euler and Hamiltonian Paths, shortest path problems, planar graphs, graph colouring, chromatic number, Euler's formula, Kuratowski's theorem. The four colour problem, Applications of Graph Colouring, Introduction to Trees, applications of trees, tree traversal, trees and sorting, Spanning trees, minimum spanning trees.

Unit 5

Languages and Grammars: Introduction to Languages and Grammars, solving problems for validity of statements according to the grammar. Importance of Language theory in Computer Science, Importance of Derivation trees, solving problems of Derivation trees, Importance of Parsing, Phrase-Structure Grammars, Types of Phrase structure grammars.

Text Book(s)

1. Kenneth H. Rosen "Discrete Mathematics and its Applications", 5th edition, McGraw-Hill Education.

Reference Material(s)

1. Kolman, Busby and Ross "Discrete Mathematical Structures", 5
th edition, Pearson Education.
2. Narsingh Deo "Graph Theory with Applications to Engineering & Computer Science", 4th
edition, Prentice Hall of India.
3. James L. Hein, “Discrete Structures, Logic and Computability”, 2
nd edition, Narosa
Publishing House.

Course Name MSc (CS/IT) 1st Semester
Subject Code: CS-4205
Subject Name: Programming and Problem-SolvingUsing

Aim of the Subject

To learn the concept of programming and enable students to develop the logical skill to solve complex problems and handle projects

Objectives

1.To develop programs to solve basic problems by implementing programming concepts like operators, control statements etc.
2.To select the right data representation formats based on the requirements of the problem.
3.To develop modular, reusable and readable C Programs using the concepts like functions, arrays etc.
4.To write programs using the Dynamic Memory Allocation concept.
5.To create, read from and write to text and binary files.

Learning Outcomes

The students are expected to be able to:
1.Formulate the algorithms for simple problems.
2.Correct syntax errors as reported by the compilers.
3.Identify and correct logical errors encountered during execution.
4.Represent and manipulate data with arrays, strings and structures.
use pointers of different types.
5. Create, read and write to and from simple text and binary files.
6.Modularize the code with functions so that they can be reused.

Unit 1

Introduction to Computer based Problem Solving; Algorithms and flowcharts; Programming Languages; Classification of Programming Languages; Characteristics of a program; Rules/conventions of coding, documentation, naming convention; Structured Programming; Modular Programming; Programming Environment: Assembler, Interpreter, Compiler, Linker and Loader.

Unit 2

Fundamentals of C programming; History of C; Structure of C Program; Character set, Identifiers and Keywords; Data types; Constants and Variables; Operators and Expressions, Type Conversion, Operator Precedence and Associativity; Basic Input/Output operations; Decision control structures :if-else, switch-case ; Loop control structure : while, do-while, for; Jump statement : break ,continue ; goto statement.

Unit 3

Array: One dimensional array -Declaration, initialization of one dimensional arrays; Two dimensional array -Declaration, initialization of two dimensional arrays; multi-dimensional array. Strings: Declaring and initializing string, reading and writing strings, string manipulation functions, array of strings. Function: Need of user-defined function, Arguments, return value, return statement; passing parameters – call by value, call by reference; Scope, visibility and lifetime of variables; Nesting of functions; passing arrays to function; passing strings to function. Recursion: basics, comparison with iteration, types of recursion. Storage Classes.

Unit 4

Pointer: Declaring and initializing pointer variables, chain of pointers, Pointer expression, Pointer arithmetic, Array of pointer and its limitations; Pointers as Function arguments; Function returning pointer, Dynamic Memory management functions. Structure: Defining a Structure, Declaring & initializing Structure Variables, Membership Operator, Array in structure, Array of Structure, Structure within structure, Pointer to structure. Union: Defining union, Declaring & initializing union Variables; Bit Fields; Enumerated data type; typedef; Bitwise operators.

Unit 5

Command line arguments; File handling: Defining, opening and closing a file, input/output operations on file, merging files; C preprocessors: Macro substitution, file inclusion, compiler control directive.

Text Book(s)

1. Herbert Schildt, “C The Complete Reference”, Osborne/McGraw-Hill, 4 th Edition, 2000.
2. Behrouz A. Forouzan and Richard F. Gilberg, “Computer Science: A Structured Programming Approach Using C”, Cengage Learning, 3rd Edition, 2007.

Reference Material(s)

1. B.W. Kernighan, D.M. Ritchie, “The C Programming Language”, Prentice Hall of India, 2nd Edition, 1988.
2. E Balagurusami, “Programming in ANSI C”, Tata McGraw-Hill, 6th Edition, 2012.
3. Byron S Gottfried, “Programming with C”, Tata McGraw-Hill, 3rd Edition, 2010.
4. Yashavant Kanetkar, “Let us C”, BPB Publications, 13th Edition, 2013.
5. Yashwant Kanetkar, “Test your C skills”, BPB Publication, 5th Edition, 2005.

Course Name MSc (CS/IT) 1st Semester
Subject Code: CS-4022
Subject Name: Computer Organization and Assembly Language Programming

Aim of the Subject

This course covers the basics of computer organisation with emphasis on the lower level abstraction of a computer system including digital logic, instruction set and assembly language programming. Topics includes data representation, logic gates, simplifi

Objectives

To understand the structure, function and characteristics of computer systems.
 To understand the design of the various functional units and components of computers.
 To identify the elements of modern instructions sets and their impact on processor design.
 To explain the function of each element of a memory hierarchy,
 To identify and compare different methods for computer I/O.

Learning Outcomes

On completion of the course, student will be able to :
 Demonstrate computer architecture concepts related to design of modern processors,
memories and I/Os.
 Analyze the performance of commercially available computers.
 To develop logic for assembly language programming

Unit 1

Computer Organization: Digital and Analog computers, Major components of a digital computer, Memory addressing capability of a CPU, Word length of a computer, Processing speed of a CPU, Definitions of Hardware, Software and Firmware. Definitions of Dumb, Smart and Intelligent terminals.
Binary Systems: Digital Systems, Binary Numbers, Number Base Conversions, Octal and Hexadecimal Numbers, Complements, Signed Binary Numbers, Binary Codes: BCD code, Gray Code, ASCII code, Excess 3 Code, Error detecting Code.

Unit 2

Computer Arithmetic: Binary representation of Negative Integers using 2's complement and Signed magnitude representation, Fixed point Arithmetic operations on Positive and Signed (Negative) Integers like addition, subtraction, multiplication, Booth algorithm for multiplication,. Division of positive and negative binary numbers.

Unit 3

Introduction of 8085 Microprocessor: Architecture of 8085 processor. Register Architecture: Accumulator, Temporally Register and Flag Register. Program Counter, Stack pointer and Instruction register. Addressing Modes: Direct addressing mode and Register direct Addressing Mode. Register Indirect Addressing Mode, Immediate Addressing Mode and Implicit or Implied Addressing Mode.

Unit 4

Introduction to Assembly Language Programming: Various Instructions Classifications: Instruction Format, Opcode, Operand and Hex code. Instruction Operation Status, Various Instruction Sets: Data Transfer Group Instructions, Arithmetic Group Instructions, Logical Group Instruction, Branch Group Instructions: Conditional and Unconditional and Machine control Instructions.

Unit 5

Assembly language programming: Practice on assembly language programming, pinout diagram of 8085 microprocessor, interfacing of 8085, interrupts, Direct memory access, introduction to 8086 microprocessor.

Text Book(s)

1. Ramesh S. Gaonkar, Microprocessor Architecture, Programming and Applications with 8085/8080. Wiley Eastern Ltd. publication
2. B Ram, Computer Fundamentals: Architecture and Organization, New Age International, 2000
3. V. Rajaraman V and N. Adabala, Fundamentals of Computers, Prentice Hall India Learning Private Limited; 6th Revised edition edition

Reference Material(s)

1. R Theagarajan S Dhanasekaran and S Dhanapal, Microprocessor and Its applications, New Age International (P) Ltd.
2. Nicholas Carter and Raj Kamal, Computer Architecture and Organization, Schaum's Outlines Series
3. Dr. Raj kamal, Digital Systems: Principles and Design, Pearson Education

Computer Organization and Assembly Language Programming

1. 8085 architecture
2. Instruction Set: Characteristics - Operand Types - Operation Types
3. Addressing Modes - instruction Formats
4. Addressing Modes (Simple Examples)
5. Assembly language programming
6. Computer Arithmetic: ALU -. Integer Representation and Arithmetic
7. Floating Point Representation and Arithmetic
8. CPU: Organization of Processors and Registers
9. Instruction Cycle - Instruction Pipelining
10. Register Optimisation - Architecture Pipelining. Instruction Set Architecture(ISA)
11. RISC and CISC, Compare RISC versus CISC
12. Characteristics of RISC, Large Register File
13. Characteristics of CISC, Instruction set complexity
14. Control Unit: Micro-Operations - Control of Processors
15. Explain how programs written in high-level languages are executed by a computer system.
16. Explain what hardware factors impact program performance and how to write programs for performance
17. Explain data representation, instruction sets, and addressing modes.
18. Write assembly language programs employing flow control constructs and procedures.
19. Explain techniques used by computer hardware designers to improve performance.
20. Explain how a data path can be implemented as a single-cycle or pipelined design.
21. Explain how the memory hierarchy impacts performance.
22. Explain the reasons for the ongoing transition to multiprocessor architectures.

CS-5511

Operating Systems

Lab Assignments

1. Implementation of FCFS (First Come First Serve) CPU Scheduling.
2. Implementation of SJF (Shortest Job First) CPU Scheduling.
3. Implementation of Round Robin (RR) CPU Scheduling.
4. Implementation of Priority CPU Scheduling Algorithm.
5. Implementation of FIFO Replacement Algorithm.
6. Implementation of Optimal Page Replacement Algorithm.
7. Implementation of LRU Page Replacement Algorithm by Stack method.
8. Implement the producer-consumer problem using threads.