Shivaji University, Kolhapur
REVISED STRUCTURE
S.E. COMPUTER SCIENCE AND ENGINEERING
W.E.F. 2014-15.

Semester - III

Sr. No.	Subject	L	T	P	Total	Theory Marks		TW	POE	Oral	Total Marks
						Written	Online				
1	Applied Mathematics	3	1	-	4	50	50	25	-	-	125
2	Discrete Mathematical Structures	4	1	-	5	50	50	25	-	-	125
3	Data Structures	4	-	-	4	50	50	-	-	-	100
4	Data Communications	4	-	-	4	50	50	-	-	-	100
5	Microprocessors	3	-	2	5	50	50	25	-	50	175
6	Programming Lab - I	2	-	4	6	-	-	50	50	-	100
7	Soft skills	-	-	2	2	-	-	50		25	75
	Total	20	2	8	30	250	250	175	50	75	800

Semester - IV

Sr. No.	Subject	L	T	P	Total	Theory Marks		TW	POE	Oral	Total Marks
						Written	Online				
1	Automata Theory	3	1	-	4	50	50	25	-	-	125
2	Computer Networks	4	-	2	6	50	50	25	-	50	175
3	Computer Organization	4	-	-	4	50	50	-	-	-	100
4	Operating System-I	3	1	-	4	50	50	25	-	-	125
5	Software Engineering	3	-	-	3	50	50	-	-	-	100
6	Programming Lab - II	2	-	4	6	-	-	50	50	-	100
7	Mini Project	-		2	2	-	-	25	-	50	75
	Total	19	2	8	29	250	250	150	50	100	800

Note:

1. The term work as prescribed in the syllabus is to be periodically and jointly assessed by a team of teachers from the concerned department.
2. In case of tutorials, students of different batches be assigned problems of different types and be guided for the solution of the problem during tutorial session. Problems thus solved be translated into computer programs wherever applicable and executed by respective batches during practical session.
3. The assignments of tutorials and practicals need to be submitted in the form of soft copy and / or written journal.

4. Breakup of term work marks shall be as follows:

    a. Mid-semester test – 5 marks.
    b. End-semester test – 5 marks.
    c. Tutorial assignments and / or practical performance – 15 marks.

5. The theory exam scheme is as under:

    a. All theory papers of SE (CSE) part 1& part 2 of 100 marks will be divided into two parts.
        i. Part-A: 50 marks theory paper similar to the existing theory paper exam. The nature of the questions will be descriptive, analytical and problem solving.
        ii. Part-B: 50 marks computer based exam with multiple choice questions (MCQs) .
    b. The marks obtained in the individual heads should be added and considered as marks of the respective theory paper out of 100 marks.
    c. The questions of part-A and part-B will be based on the entire syllabus of the respective subjects.
    d. The questions in part-B will be of 1 or 2 marks only.
    e. Duration of part-A exam will of 2 hours and that of part-B will be of 1 hour.
    f. The passing scheme for the subject will be similar to existing scheme.
    g. No separate passing head for part-A and part-B.
    h. The scheme of revaluation is not applicable for part-B, however is applicable for part-A
    i. All the existing ordinances will be applicable for passing criteria.

S.E. (Computer Science and Engineering) Semester – III

APPLIED MATHEMATICS

Lectures : 3 hours/week **Theory : 100 marks**
Tutorial : 1 hour/week **Term Work : 25 marks**

Course Objectives:

1) To teach mathematical methodologies and models to develop mathematical skills and enhance thinking power of students.
2) To expose the students to the topics in fuzzy set theory, numerical methods and statistics with an emphasis on the application of solving engineering problems.
3) To prepare students to formulate a mathematical model using engineering skills & interpret the solution in real world.

Unit 1. Curve Fitting: **(6)**

1.1 Lines of regression of bivariate data,
1.2 Fitting of Curves by method of Least-squares.
- 1.2.1 Fitting of Straight lines
- 1.2.2 Fitting of exponential curves.
- 1.2.3 Fitting of Parabolic curves.

Unit 2. Probability Distribution: **(6)**

2.1 Random variable
2.2 Binomial Distribution
2.3 Poisson Distribution
2.4 Normal Distribution

Unit 3. Numerical solution of transcendental & algebraic equations and Numerical Integration **(6)**

3.1 Newton Raphson Method
3.2 Secant method
3.3 Trapezoidal Rule
3.4 Simpson's 1/3 rd rule
3.5 Simpson's 3/8 th rule

Unit 4. Introduction to Fuzzy sets: **(6)**

4.1 Basic concepts of fuzzy sets
4.2 Crisp set and Fuzzy set.
4.3 Membership functions
4.4 Basic operations on fuzzy sets
4.5 Properties of fuzzy sets

Unit 5. Fuzzy Arithmetic: **(6)**

5.1 Fuzzy numbers
5.2 Fuzzy cardinality
5.3 Operations on Fuzzy numbers
5.4 Fuzzy equations of type A + X = B and A.X = B

Unit 6. Assignment Problems: **(6)**

6.1 Definition, Balanced and Unbalanced assignment problems,
6.2 Hungarian method of solving balanced assignment problems.
6.3 Hungarian method of solving unbalanced assignment problems.
6.4 Traveling salesmen problem.

General Instructions:

1. For the term work of 25 marks, batch wise tutorials are to be conducted. The number of students per batch should be 20 i.e. as per University pattern for practical batches.
2. Minimum number of assignments should be 8 covering all topics.

Reference Books:

1. A text book of Applied Mathematics: Vol. I, II and III by J. N. Wartikar & P. N. Wartikar Vidyarthi Griha Prakashan, Pune.
2. Higher Engineering Mathematics by Dr. B. S. Grewal.
3. Operations Research by S. D. Sharma
4. Fuzzy sets and Fuzzy Logic by George J. Klir, Bo Yuan.
5. Probability and Statistics for Computer science by James L. Johnon (Wiley Student Edition).

DISCRETE MATHEMATICAL STRUCTURES

Lectures: 4 hrs / week **Theory: 100 marks**
Tutorial: 1 hr / week **Term work: 25 marks**

Course Objectives:

1. To expose the students to the mathematical logic related to Computer science areas.

2. To enhance the problem solving skills in the areas of theoretical computer science.

3. To use the mathematical concepts in the development of computer applications..

Unit 1 : Mathematical logic: (9)

Introduction, statements and notations, connectives – negation, Conjunction, disjunction, conditional, bi-conditional, Statement formulas and truth tables, well formed formulas, Tautologies, Equivalence of formulas, Duality law, Tautological implications, functionally complete sets of connectives, other connectives, Normal and principal normal forms, completely parenthesized infix and polish notations, Theory of Inference for statement calculus – validity using truth table, rules of inference, consistency of Premises and indirect method of proof.

Unit 2 : Set theory (10)

Basic concepts of set theory, types of operations on sets, ordered pairs, Cartesian Product, representation of discrete structures, relation, properties of binary relations, matrix and graph representation, partition and covering of set, equivalence relation, composition, POSET and Hasse diagram, Function – types, composition of functions, Inverse function.

Unit 3 : Algebraic systems (5)

Semigroups and Monoids, properties and examples, Groups: Definition and examples, subgroups and homomorphism.

Unit 4 : Lattices and Boolean algebra (8)

Lattice as POSETs , definition , examples and properties, Lattice as algebraic systems, Special lattices, Boolean algebra definition and examples, Boolean functions, representation and minimization of Boolean functions.

Unit 5 : Graph theory (5)

Basic concepts of graph theory, Storage representation and manipulation of Graphs, PERT and related techniques.

Unit 6: Permutations, Combinations and Discrete Probability (8)

Permutations and Combinations: rule of sum and product, Permutations, Combinations, Algorithms for generation of Permutations and Combinations. Discrete Probability, Conditional Probability, Bayes' Theorem, Information and Mutual Information.

Text Books:

1. Discrete Mathematical Structures with Application to Computer Science - J. P. Tremblay & R. Manohar (MGH International)
2. C. L. Liu and D. P. Mohapatra, "Elements of Discrete Mathematics", SiE Edition, TataMcGraw-Hill, 2008,ISBN 10:0-07-066913-9 (For Unit no 6)

References:

1. Discrete Mathematics - Semyour Lipschutz, MarcLipson (MGH), Schaum's outlines.
2. Discrete Mathematics and its Applications - Kenneth H. Rosen (AT&T Bell Labs) (mhhe.com/rosen)
3. Schaums Solved Problem Series – Lipschutz.
4. Discrete Mathematical Structures – Bernard Kolman, Robert Busby, S.C.Ross and Nadeemur-Rehman (Pearson Education).

Term Work :

It should consist of minimum 10 to 12 assignments based on following guidelines. In tutorial session, students of different batches be assigned –

a) Different exercise problems and be guided for the solution of the problems AND

b) To write programs in C language on any 4 to 5 following related topics.

1. Generating truth table of a statement
2. Application of bit representation of sets and operations on sets or relations.
3. Conversion of polish expressions.
4. Obtaining the path matrix, paths of different lengths.
5. Different tree traversal methods.
6. Evaluating polynomial expressions using linked lists.
7. Allocation graphs and deadlock detection.
8. PERT related techniques.
9. Algorithms for generation of Permutations & Combinations.

DATA STRUCTURES

Lectures: 4 hrs/week **Theory: 100 marks**

Course Objectives:

1. To make the students familiar with basic data structures.
2. To teach the students to select appropriate data structures in computer applications.
3. To provide the students with the details of implementation of various data structures.

Unit 1: Basics of Data Structures: **(6)**

Overview of C- Basic data types, control structures, array, function, structure, pointers, Time and Space complexity.

Unit 2: Searching and Sorting Techniques: **(10)**

Linear search, binary search, bubble sort, selection sort, insertion sort, merge sort, quick sort, radix sort, heap sort

Hashing – Definition, hash functions, overflow, collision, open and closed hashing, rehashing techniques.

Unit 3: Stacks and Queues: **(7)**

Stacks: Definition, representation, operations, static implementation and applications of stack. **Queues:** Definition, representation, operations, static implementation and applications of queue, circular queue, priority queue.

Unit 4: Lists: **(10)**

Definition representation, operations, implementation and applications of singly, doubly and circular linked lists. Implementation of stack and queue using linked lists.

Unit 5: Trees: **(7)**

Basic terminology, representation, binary tree, traversal methods, binary search tree, AVL search tree, B tree, B+ tree, Heaps- Operations and their applications.

Unit 6: Graphs: **(6)**

Basic concept of graph theory, storage representation, graph traversal techniques- BFS and DFS, Graph representation using sparse matrix.

Text Books:

1. Let us C – Yashwant Kanetkar (BPB) (Chapter 1)
2. Schaum's Outlines Data Structures – Seymour Lipschutz (MGH) (Chapter 2 to 6)

Reference books:

1. Data Structure using C- A. M. Tanenbaum, Y. Langsam, M. J. Augenstein (PHI)

2. Data Structures- A Pseudocode Approach with C – Richard F. Gilberg and Behrouz A. Forouzon 2nd Edition

DATA COMMUNICATIONS

Lectures: 4 hrs/ week **Theory: 100 Marks**

Course Objectives:

1. To learn basic concepts and principles of data communication.
2. To expose the students to various Computer Network models and standards.
3. To expose the students to IEEE standards for LAN.

Unit 1:- Introduction

1. Introduction : Data communications, Networks, Protocols & standards (3)
2. Network Models:- Layered Tasks, The OSI model, Layers in the OSI model, TCP/IP protocol suit, ATM reference model. (4)

Unit 2:- Communication Basics

3. Data & Signals :- Analog & Digital, Periodic analog signals, digital signals, Transmission Impairments, Data rate limits & Performance (5)
4. Digital Transmission :- Line coding & line coding schemes (Unipolar, polar & bipolar) Transmission models (3)

Unit3 :- Physical Layer

5. Transmission media :- Guided, Unguided media (4)
6. Network Hardware components:-
   Transceivers & media converters, Repeaters, NIC & PC cards, Bridges, switches, Routers (4)

Unit 4 : Data Link Layer :-

7. Error detection & correction: Block coding, cyclic codes, checksum (3)
8. Data Link Control :- Framming, Flow & error control, stop & wait protocol, sliding window protocol, HDLC protocol. (5)

Unit 5 :- The medium Access control (7)

9. Channel allocation Problem, Multiple Access Protocols, ALHOA, CSMA, collision free protocols, Limited contention protocols.

Unit 6 :- IEEE standards For LANS & MANS :- (8)

10. 802.3 Standard & Ethernet,
    802.4 Standard & Token Bus,
    802.5 Standard & Token Ring,
    Comparison of 802.3, 802.4 and 802.5,
    802.6 standard (DQDB) and 802.2 logical link control.

Text Books:-

1. Data Communications and Networking – Behrouz A Forouzan (The McGraw Hill) (ch : 1,2,3,4,5,7,8).
2. Computer Networks – Andrew S. Tanenbaum- (Prentice Hall) (ch:9,10,11) 5th Edition.
3. Computer communications and Networking Technologies – Michael A Gallo (Cengage Learning) (ch:6).

Reference Books :-

1. Data & computer communications :- William Stallings (Pearson Education).
2. Data communication and computer Networks Ajit Pal (PHI Learning) .

MICROPROCESSORS

Lectures : 3 hrs/ week **Theory: 100 Marks**
Practicals: 2 hrs/ week **Term work : 25 Marks**
Oral Exam : 50 Marks

Course Objectives:

1. To expose the students to architectures of basic microprocessors.
2. To give the hands on experience on assembly language programming for microprocessors.

Unit 1: Intel 8085 Architecture & Programming (6)

1.1 Architecture of 8085
1.2 Instruction set & Execution in 8085
1.3 Classification of Instructions
1.4 Instruction set of 8085
1.5 Sample Programs
1.6 Assembler
1.7 Assembly Language Programs

Unit 2: Microprocessor and Architecture (6)

2.1 The Microprocessor Based Personal Computer System.
2.2 Internal Microprocessor Architecture
2.3 Real Mode Memory Addressing
2.4 Introduction to Protected Mode memory Addressing
2.5 Memory Paging

Unit 3: Addressing Modes and Data Movement Instructions (6)

3.1 Data Addressing Modes
3.2 Program Memory Addressing Mode
3.3 Stack Memory Addressing Mode
3.4 MOV Revisited
3.5 PUSH/POP
3.6 Load Effective Address
3.7 String Data Transfer
3.8 Miscellaneous Data Transfer Instruction
3.9 Segment Override Prefix
3.10 Assembler Details

Unit 4: Arithmetic, Logic & Program Control Instructions (6)

4.1 Addition ,Subtraction and Comparison
4.2 Multiplication and Division

4.3 BCD and ASCII Arithmetic
4.4 Basic Logic Instructions
4.5 Shift and Rotate
4.6 The Jump Group
4.7 Controlling the Flow of program
4.8 Machine control and Miscellaneous Instructions

Unit 5: Interrupts and The 80386 Microprocessor (6)

5.1 Basic Interrupt Processing
5.2 Hardware Interrupt
5.3 The80386 Microprocessor: The memory System
5.4 Special 80386 Registers
5.5 80386 Memory Management
5.6 Virtual 8086 Mode
5.7 The Memory Paging Mechanism

Unit 6: Pentium , Pentium Pro and Pentium 4 Microprocessor (4)

6.1 The Pentium Microprocessor : The Memory System
6.2 Special Pentium Registers
6.3 Pentium Memory Management
6.4 The Pentium Pro Microprocessor : Internal structure of the Pentium Pro
6.5 The Pentium 4 : Memory Interface, Register Set, Hyper Threading Technology CPUID

Text Books:

1) The INTEL Microprocessors - Architecture ,Programming and Interfacing - Barry B. Brey Seventh Edition (PHI Ltd)
2) Microprocessors and Microcontrollers - N. Senthi Kumar, M. Saravanan and S. Jeevananthan (Oxford University Press)

Reference Books:

1) Microprocessor Architecture, Programming and Applications with 8085 - Ramesh Gaonkar
2) The Microcomputer systems: The 8086/8088 Family - Yu Cheng Liu , Glenn A. Gibson (PHI Ltd)

Term Work :

It should consists of 10 to 12 experiments based on the following guidelines.

a) Experiment No 1 to 6 should be based on 8085 Instruction set – out of which :

Two experiments on : Different 5 to 6 Programs using Data Transfer Instructions.

Two experiments on : Different 5 to 6 Programs using Arithmetic & logic Instructions.
Two experiments on : Different 3 to 4 Programs using Branching Instructions.

b) Experiment No 7 to 12 should be based on 8086, 80286 & 80386 Instruction set using Assembler out of which :

Two experiments on: Different 4 to 5 small Assembly Programs using Addressing Modes and Data Movement Instructions.

Two experiments on: Different 4 to 5 small Assembly Programs using ARTHMETIC, LOGIC & PROGRAM CONTROL INSTRUCTION.

Two experiments on: Different 4 to 5 small Assembly Programs using Bios Interrupts in High Level Language C.

PROGRAMMING LABORATORY- I

Lectures : 2 hrs / week **Term work : 50 marks**
Practical : 4 hrs / week **POE : 50 marks**

Course Objectives:

1. To expose the students to programming constructs of C language.
2. To learn the implementation of various data structures using C.
3. To learn how to write modular and efficient C programs.

Unit 1: An Overview of C : 5

Compilers vs. Interpreters, The Form of a C Program, The Library and Linking, Separate Compilation, Compiling a C Program, C's Memory Map; Expressions – The Basic Data Types, Modifying the Basic Types, Identifies Names, Variables, The Four C Scopes, Type Qualifiers-const, volatile, Storage Class Specifiers; Statements - Selection Statements, Iteration Statements, Jump Statements, Expression Statements, Block Statements.

Console I/O: Reading and Writing Characters, Reading and Writing Strings, Formatted Console I/O, printf(), scanf(), Suppressing Input.

The Preprocessor and Comments
The Preprocessor, #define, #error, #include, Conditional Compilation Directives, #Undef, Using defined, #line

Unit 2: Arrays : 3

Arrays and Strings- Two-Dimensional Arrays, Arrays of Strings, Multidimensional Arrays, Array Initialization, Variable-Length Arrays.

Unit 3: Functions : 4

The General Form of a Function, Understanding the Scope of a Function, Parameter passing, Passing arrays to functions, Function Arguments, argc and argv-Arguments to main(),The return Statement, What Does main( ) Return?, Recursion, Function Prototypes, Declaring Variable Length Parameter Lists, The "Implicit int" Rule, Old-Style vs. Modem Function Parameter Declarations, The inline Keyword.

Unit 4: Pointers : 5

What Are Pointers?, Pointer Variables, The Pointer 0perators, Pointer Expressions, Pointers and Arrays, Arrays of Pointers, Multiple Indirection, Initializing Pointers, Pointers to Functions and structures, C's Dynamic Allocation Functions, restrict-Qualified Pointers, Problems with Pointers.

Unit 5: Structures, Unions, Enumerations, and typedef : 4

Structures, Arrays of Structures, Passing Structures to Functions, Structure Pointers, Arrays and Structures Within Structures, Unions, Bit-Fields, Enumerations, Using sizeof to Ensure Portability, typedef .

Unit 6: File I/O : 3

File I/O, Standard C vs. Unix File I/O, Streams and Files, File System Basics, fread( ) and fwrite(), fseek() and Random-Access I/O, fprintf( ) and fscanf(), The Standard Streams.

Text Books:

1. Let Us C- Yashvant Kanetkar (BPB Publications)
2. Understanding Pointers in C- Yashvant Kanetkar (BPB Publications)

Reference Books:

1) C The Complete Reference – Herbert Schildt (Tata McGraw-Hill Edition)
2) The C Programming Language- Brian W. Kernighan, Dennis Ritchie (Prentice Hall Software Series)

Term Work:

It should consist of minimum 14 experiments based on the following guidelines and should be conducted in Unix/Linux platform.

1. Implement matrix operation by representing matrix in the form of (a) array (b) linked list. Matrix

   Operation like – Multiplication of matrices, finding the Inverse matrix, singular matrix, upper triangle, lower triangle, symmetric matrix, skew symmetric, triangular matrix etc

2. Implement a C program that will accept a hexadecimal number as input and then display a menu that will permit any of the following operation to be carried out.

(a) Display hex equivalent of one's complement.

(b) Carry out a masking operation and then display the hex equivalent of the result.

(c) Carry out a bit shifting operation and then display the hex equivalent of the result.

(d) Exit.

If the masking operation is selected, prompt the user for the type of operation (bit wise and bit wise exclusive or bit wise or) and then a (hex) value for the mask. If the shifting operation is selected, prompt the type of shift (left or right) and then the no. of bits to be shifted. Test the program with several different (hex) input values of your own choice. Note: Conversion of different equivalent forms like – binary, octal, decimal and others can also be performed and tested.

3. Define a mask and write the appropriate masking operating for each of the situations described below:

   a. Copy the odd bits (bits 1, 3, 5…….15) and place zero in the even-bit location (bit 0, 2, 4, 14) of 16 bit, unsigned integer quantity represented by the variable v. Assume that bit 0 is the rightmost bit.

   b. Strip the msb (the leftmost bit) from an 8-bit character represented by variable c (certain word processor use this bit to control the formatting of the text within a document. Stripping this bit

      i. e. setting it to zero, can transform the word processor documents into a text file consisting of ordinary ASCII characters.)

   c. Copy the odd bits (bits 1, 3, 5,…., 15) and place one's in the even bit locations (bits 0,2,4,…..,14) of a 16 bit unsigned integer quantity represented by variable v. Assume bit 0 is the rightmost bit.

   d. Toggle (invert) the values of bits 1 & 6 of the 16 bit unsigned integer quantity represented by variable v, while preserving all the remaining bits. Assign new bit pattern to v.

4. Implement and compare linear and binary search for any given huge data set <min – 10000>. Data must be float, string.
5. Implement any 2-3 sorting techniques and find the number of comparison required to sort random data –set of around 10000.
6. Write a problem which **a.** Implements pointers to function **b.** has a function having parameters as pointer to function.
7. Implement stack using array and linked list.
8. Implement queue, priority queue, circular queue using array and linked list.
9. Write a program which shows advantages of **(a)** Static variable **(b)** Static function (using multiple C files) **(c)** volatile **(d)** Extern (using multiply C files.)
10. Implement hashing and rehashing operating on data like float and string.
11. Implement doubly linked list, Circular linked list, doubly circular linked list.

12. Implement and perform different operation of binary tree, B-tree insertion, deletion, modification, finding the depth of the tree.
13. Create your own library file and header file.
14. Implement all loops. Also implement equivalent loops –for, while, do-while using recursion.
15. Implement Towers of Hanoi and Ackermann's function.

Breakup of term work marks:

Mid-semester **Practical test** – 10 marks.
End-semester **Practical test** – 10 marks.
Practical performance – 30 marks.

SOFT SKILLS

Practicals: 2 hrs /week Term work: 50 marks Oral Exam: 25 marks

Objectives:

1. To enhance the communications skills of the students.
2. To expose the students to basic skills of team work.
3. To inculcate the writing skills necessary for business communications.

Unit I: Communication Skills

Verbal Communication - Effective Communication - Active listening – Articulation- Paraphrasing – Feedback

Non Verbal Communication - Body Language of self and others

Importance of feelings in communication - dealing with feelings in communication Inter and Intrapersonal communication- Self-esteem and confidence - Assertiveness

Unit II: Importance of Team work

Self Enhancement - importance of developing assertive skills- developing self-confidence – developing emotional intelligence.

Importance of Team work – Team vs. Group - Attributes of a successful team – Barriers involved

Working with Groups – Dealing with People- Group Decision Making
Effective teams – Elements of Team work - Stages in team formation

Unit III: Writing

Introduction to writing, Hallmark of good writing, Writing conventions, business writing, writing a notice, writing styles, e-mail writing, report writing, practice.

Text Books:

1. Developing Communication Skills by Krishna Mohan and Meera Banerji; MacMillan India Ltd., Delhi
2. Essentials of Effective Communication, Ludlow and Panthon; Prentice Hall of India.
3. Seven Spiritual Laws of Success - Deepak Chopra.
4. Good To Great - Jim Collins.