University Institute of Technology (UIT)
Silver Wood Estate, H. P. University Shimla-171005
(NAAC accredited "A"Grade University)

Revised Syllabus
Bachelor of Technology

In
Computer Science Engineering

Effective for batch 2019-2020 and onwards

Scheme for Syllabus of B.Tech (CSE) II year
Semester III

SN	Cat.	Code	Course Title	Hours Per week			Credits	Marks	
				L	T	P	C	Ext.	Int.
1	ESC	ES-3001	Discrete Mathematics	3	1	0	3	100	50
2	PCC	CS-3001	Data Structures	3	1	0	3	100	50
3	PCC	CS-3002	Computer Organization & Architecture	3	1	0	3	100	50
4	ESC	CS-3003	Object Oriented Paradigm	3	1	0	3	100	50
5	PCC	ECE-3003	Digital Electronics	3	1	0	3	100	50
6	HSMC	HS-3001	Principle of Engineering Economics and Management	3	1	0	2	100	50
7	PCC	CS-3051	Data Structures Laboratory with C/C++	0	0	2	1	50	50
8	PCC	CS-3052	Object Oriented Programming Laboratory with C++	0	0	2	1	50	50
9	PCC	ECE-3053	Digital Electronics Laboratory	0	0	2	1	50	50
	Total			**30**			**20**	**1200**	

Semester IV

SN	Cat.	Code	Course Title	Hours Per week			Credits	Marks	
				L	T	P	C	Ext.	Int.
1	ESC	ES-4001	Numerical Method	3	1	0	3	100	50
2	PCC	CS-4001	Operating System	3	1	0	3	100	50
3	PCC	CS-4002	Software Engineering	3	1	0	3	100	50
4	PCC	CS-4003	Analysis and Design of Algorithm	3	1	0	3	100	50
5	PCC	CS-4004	Theory of Computation	3	1	0	3	100	50
6	PCC	CS-4005	Python Programming	3	1	0	3	100	50
7	PCC	CS-4051	Operating System Laboratory	0	0	2	1	50	50
8	PCC	CS-4052	Analysis and Design of Algorithm Laboratory	0	0	2	1	50	50
9	PCC	CS-4053	Python Programming Laboratory	0	0	2	1	50	50
10	PCC	CS-4054	Software Engineering Project	0	0	2	1	50	50
	Total			**32**			**22**	**1300**	

L	**–**	**No of lectures per week**	**T**	**–**	**No of tutorials per week**
P	**–**	**No of practical per week**	**C**	**–**	**Credits**
Ext.	**--**	**External Marks**	**Int.**	**--**	**Internal Marks**
Cat	–	**Course category**			

Course Category and Definition:

Course Category	Definitions
BSC	Basic Science Courses
ESC	Engineering Science Courses
HSMC	Humanities and Social Sciences including Management courses
PCC	Professional core courses

DETAILED 4-YEAR CURRICULUM CONTENTS
Undergraduate Degree in Engineering ∓ Technology

Branch/Course: COMPUTER SCIENCE AND ENGINEERING

Second year (Third semester onwards)

PROFESSIONAL CORE COURSES

THIRD SEMESTER

Discrete Mathematics (ES - 3001)

Course Code	**ES - 3001**	**Credits-3**	**L-3, T-1, P-0**
Name of the Course	**Discrete Mathematics**		
Lectures to be Delivered	**52 (1 Hr Each) (L=52 for each semester)**		
Semester End Examination	**Max Marks: 100**	**Min Pass Marks: 40**	**Maximum Time: 3 hrs**
Continuous Assessment (based on sessional tests (2) 50%, Tutorials/Assignments 30%, Quiz/Seminar 10%, Attendance 10%)			**Max Marks: 50**

Instructions

1. **For Paper Setters:** The question paper will consist of five sections A, B, C, D & E. Section E will be compulsory, it will consist of a single question with 10-20 subparts of short answer type, which will cover the entire syllabus and will carry 20% of the total marks of the semester end examination for the course. Section A, B, C & D will have two questions from the respective sections of the syllabus and each question will carry 20% of the total marks of the semester end examination for the course.
2. **For candidates:** Candidates are required to attempt five questions in all selecting one question from each of the sections A, B, C & D of the question paper and all the subparts of the questions in Section E. Use of non-programmable calculators is allowed.

Course Description:

Section A

Mathematic Logic: Statements and Notation, Connectives; Negation; Conjunction; Disjunction; Statement Formulas and Truth Tables; Logical Capabilities of programming Languages; Conditional and Biconditional; Well formed Formulas; Tautologies; Equivalence of Formulas; Duality Law; Tautological Implications; Formulas with distinct Truth Tables; Functionally Complete sets of connectives; other connectives; Two state devices and Statement logic; Normal forms; principal disjunctive normal forms; principal conjunctive normal form; ordering and Uniqueness of Normal Forms; Completely parenthesized Infix Notation and Polish Notation; The Theory of Interference for the statement calculus; Validity using Truth Table; Rules of Inference; Consistency of Premises and Indirect Method Of Proof; Automatic Theory Proving; The Predicate Calculus; predicates; The Statement Function, variables and Quantifiers; Predicate Formulas; Free and Bound Variables; The Universe of Discourse; Inference Theory of the Predicate Calculus, Valid Formulas and Equivalences; Some Valid Formulas over Finite Universes; Special Valid Formulas Involving Quantifiers; Theory of Inference for the Predicate Calculus; Formula Involving More Than One Quantifier.

Section B

Permutations, Combinations, and Discrete Probability: Introduction, The Rules of Sum and product; permutations; Combinations; Generation of permutations and

combinations, Discrete probability, Information and Mutual Information.
Relations and Functions: Introduction, A Relational Model for Data Bases; properties of Binary Relations; Equivalence Relations and partitions; Partial Ordering Relations and Lattices; Chains and Antichains; A Job Scheduling problem; Functions and the Pigeonhole principle.

Section C

Graphs and Planner Graphs: Introduction, Basic Terminology, Multigraphs and Weighted Graphs, Paths and Circuits; Shortest paths in Weighted Graphs, Eulerian paths and circuits; Hamiltonian paths and circuits, The Traveling Salesperson problem; Factors of Graph; planar Graph.
Trees and cut-sets: Trees, Rooted Trees, path, Lengths in Rooted trees; prefix codes; Binary search trees; Spanning Trees and cut-sets; Minimum Spanning Trees; Transport Networks.

Section D

Recurrence Relations and Recursive Algorithms: Introduction; Recurrence Relations; Linear Recurrence Relations with constant coefficients; Homogeneous Solutions; Particular Solutions; total Solutions; solution by the Method of Generating Functions; Sorting Algorithms; Matrix Multiplication Algorithms.
Groups and Rings: Introduction, Groups, Subgroups; Generators and evaluation of Poers; Cosets and Lagrange's Theorem; permutation groups and Burnside's theorem; Codes and Group codes; Isomorphisms and Automorphisms; Homomorphisms and Normal Subgroups; Rings, Integral Domains, and Fields; Fing Homomorphisms; Polynomial Rings and Cyclic Codes.

Books:

1. J.P. Trembley and R. Manohar, "Discrete mathematics Structures with Applications to Computer Science", (TaTa McGraw-Hill, 1997)
2. C.L.Liu, " Elements of Discrete Mathematics", 2nd Edition (TaTa McGraw-Hill, 1985)
3. Kenneth H. Rosen, *Discrete Mathematics and Its Applications*, McGraw-Hill Education, Eighth Edition, 2019
4. Harry Lewis, & Rachel Zax, *Essential Discrete Mathematics for Computer Science*, Princeton University Press, 2019.

Data Structures (CS - 3001)

Course Code	**CS - 3001**	**Credits-3**	**L-3, T-1, P-0**
Name of the Course	**Data Structures**		
Lectures to be Delivered	**52 (1 Hr Each) (L=39, T=13 for each semester)**		
Semester End Examination	**Max Marks: 100**	**Min Pass Marks: 40**	**Maximum Time: 3 hrs**
Continuous Assessment (based on sessional tests (2) 50%, Tutorials/Assignments 30%, Quiz/Seminar 10%, Attendance 10%)			**Max Marks: 50**

Instructions

1. **For Paper Setters:** The question paper will consist of five sections A, B, C, D & E. Section E will be compulsory, it will consist of a single question with 10-20 subparts of short answer type, which will cover the entire syllabus and will carry 20% of the total marks of the semester end examination for the course. Section A, B, C & D will have two questions from the respective sections of the syllabus and each question will carry 20% of the total marks of the semester end examination for the course.
2. **For candidates:** Candidates are required to attempt five questions in all selecting one question from each of the sections A, B, C & D of the question paper and all the subparts of the questions in Section E. Use of non- programmable calculators is allowed.

Course Description:

Section A

Data types, data structures, abstract data types, the running time of a program, the running time and storage cost of algorithms, complexity, asymptotic complexity, big O notation, obtaining the complexity of an algorithm. Notations and Analysis, Storage structures for arrays - sparse matrices - structures and arrays of structures, Stacks and Queues: Representations, implementations and applications.

Section B

Linked List

Singly linked lists, Linked stacks and queues, operations on Polynomials, Doubly Linked Lists, Circularly Linked Lists, Operations on linked lists- Insertion, deletion and traversal, dynamic storage management – Garbage collection and compaction.

Trees

Basic terminology, General Trees, Binary Trees, Tree Traversing: inorder, preorder and postorder traversal, building a binary search tree, Operations on Binary Trees - Expression Manipulations - Symbol Table construction, Height Balanced Trees(AVL), B-trees, B+ -trees.

Section C

Graph

Basic definitions, representations of directed and undirected graphs, the single-source shortests path problem, the all-pair shortest path problem, traversals of directed and undirected graphs, directed acyclic graphs, strong components, minimum cost spanning tress, articulation points and biconnected components, graph matching.

Section D

Sorting and Searching Techniques

Bubble sorting, Insertion sort, Selection sort, Shell sort, Merge sort, Heap and Heap sort, Quick sort, Radix sort and Bucket sort, Address calculation, Sequential searching, Binary Searching, Index searching, Hash table methods.

Text and Reference Books

1. Seymour Lipschutz "Data Structures" Schaum's ouTlines (Revised edition)McGraw Hill Education Pvt.Ltd. New Delhi
2. J.P. Tremblay and P.G. Sorenson, "An Introduction to Data Structures with applications", Tata McGraw Hill.
3. S.Sahni, "Data structures, Algorithms ad Applications in C++", WCB/McGraw Hill.
4. Aho ,Ullman and Hopcroft, " Data Structures and Algorithms".
5. Y. Langsam, M. J. Augenstein and A. M. Tenenbaum, "Data Structures using C", Pearson Education
6. Richard F. Gilberg, Behrouz A. Forouzan, "Data Structures – A Pseudocode Approach with C", Thomson Brooks / COLE

Computer Organization & Architecture (CS- 3002)

Course Code	**CS - 3002**	**Credits-3**	**L-3, T-1, P-0**
Name of the Course	**Computer Organization & Architecture**		
Lectures to be Delivered	**52 (1 Hr Each) (L=39, T=13 for each semester)**		
Semester End Examination	**Max Marks: 100**	**Min Pass Marks: 40**	**Maximum Time: 3 hrs**
Continuous Assessment (based on sessional tests (2) 50%, Tutorials/Assignments 30%, Quiz/Seminar 10%, Attendance 10%)			**Max Marks: 50**

Instructions

1. **For Paper Setters:** The question paper will consist of five sections A, B, C, D & E. Section E will be compulsory, it will consist of a single question with 10-20 subparts of short answer type, which will cover the entire syllabus and will carry 20% of the total marks of the semester end examination for the course. Section A, B, C & D will have two questions from the respective sections of the syllabus and each question will carry 20% of the total marks of the semester end examination for the course.
2. **For candidates:** Candidates are required to attempt five questions in all selecting one question from each of the sections A, B, C & D of the question paper and all the subparts of the questions in Section E. Use of non- programmable calculators is allowed.

Course Description:

Section-A

BASICS: An introduction to computers with block diagram, Computers generation, Impact of technology. LOGIC DESIGN TECHNIQUES: Designing combinations logic using Karnaugh-Maps with building blocks of basic gates , Multiplexers, de-multiplexer, decoders and encoders, arithmetic, logics units, Instruction codes, Computers registers and instructions, timing and control, instruction cycle, memory reference instruction, I –O interruption, Basic sequential logic blocks flip-flops, registers, shift registers and counters, Finite state Machine using state tables

Section-B

COMPUTER ARITHMETIC: Adder and Subtractor circuits, Booth Multiplication algorithm, Performance bench marks. CONTROL PATH DESIGN: Sequence counter method, Micro programmed controllers, address sequencing, symbolic micro –instructions

CENTRAL PROCESSING UNIT: Registers general register origination, stack origination, Instruction formats, address instructions, addressing modes, data transfer and manipulations, programmed control RISC instruction set design, three address instructions and arithmetic pipelines with example of floating point adder, instruction pipe lines, advanced pipe lining using instruction level parallelism

Section-C:

The Processor: Datapath and Control: Introduction, Building a Datapath for Supporting the ISA, Single Cycle Implementation, Multi Cycle Implementation, Exceptions, Micro-programming, Hard-wired Control

Enhancing Performance with Pipelining: An Overview of Pipelining, Pipelined Datapath, Pipelined Control, Data Hazards and Forwarding, Data Hazards and Stalls, Control Hazards, Exception Handling.

Instruction Level Parallelism and its Exploitation: Instruction Level Parallelism: Concepts and Challenges, Basic Compiler Techniques for Exposing ILP, Reducing Branch Cost with Speculation, Overcoming Data Hazards with Dynamic Scheduling, Exploiting ILP Using Multiple

Issue and Scheduling, Advanced Techniques for Instruction Delivery and Speculation.
Assessing and understanding Performance: Introduction, CPU Performance and its Factors, Evaluating Performance.

Section –D

Instruction Set Principles and Examples (example of MIPS): Introduction, Classifying Instruction Set Architectures, Memory Addressing, Type and Size of Operands, Operations in the Instruction Set, Instructions for Control Flow, Encoding an Instruction Set, Role of Compilers, MIPS Instruction Set Architecture.

Caches and Memory Hierarchy Design: Introduction, the Basics of Caches, Measuring and Improving Cache Performance, Basic Cache Optimizations, Virtual Memory, Memory Hierarchies, Scratch pad Memories.

Text and Reference Books

1. M. Moris Mano , Computer System &Architecture PHI
2. Hayes J. P Computer Architecture & Organization. William C Brown Publisher
3. David A Patterson & John L Hennessy, "Computer Organization & Design: A Harware/Software Interface", Morgan Kaufmann Publishers.
4. John L Hennessy & David A Patterson, "Computer Architecture: A Quantitative Approach", Morgan Kaufmann Publishers.

Object Oriented Paradigm (CS - 3003)

Course Code	**CS - 3003**	**Credits-3**	**L-3, T-0, P-0**
Name of the Course	**Object Oriented Paradigm**		
Lectures to be Delivered	**52 (1 Hr Each) (L=52 for each semester)**		
Semester End Examination	**Max Marks: 100**	**Min Pass Marks: 40**	**Maximum Time: 3 hrs**
Continuous Assessment (based on sessional tests (2) 50%, Tutorials/Assignments 30%, Quiz/Seminar 10%, Attendance 10%)			**Max Marks: 50**

Instructions

1. **For Paper Setters:** The question paper will consist of five sections A, B, C, D & E. Section E will be compulsory, it will consist of a single question with 10-20 subparts of short answer type, which will cover the entire syllabus and will carry 20% of the total marks of the semester end examination for the course. Section A, B, C & D will have two questions from the respective sections of the syllabus and each question will carry 20% of the total marks of the semester end examination for the course.
2. **For candidates:** Candidates are required to attempt five questions in all selecting one question from each of the sections A, B, C & D of the question paper and all the subparts of the questions in Section E. Use of non- programmable calculators is allowed.

Course Description:

Section-A

Concepts of Object-Oriented Programming

Oriented Programming Paradigm, Basic concepts of OOP's, Benefits of OOPS, Introduction to object oriented design and development, Design steps, Design example, Object oriented languages, Comparison of structured and object-oriented programming languages.

Arrays, Pointers and Functions

Arrays, Storage of arrays in memory, Initializing Arrays, Multi-Dimensional Arrays, Pointers, accessing array elements through pointers, Passing pointers as function arguments, Arrays of pointers, Pointers to pointers, Functions, Arguments, Inline functions, Function Overloading Polymorphism.

Section - B

Classes and Objects

Data types, operators, expressions, control structures, arrays, strings, Classes and objects, access specifiers, constructors, destructors, operator overloading, type conversion.

Storage classes

Fixed vs Automatic declaration, Scope, Global variables, register specifier, Dynamic memory allocation.

Inheritance

Inheritance, single Inheritance, Multiple Inheritance, Multi-le vel inheritance, hierarchical inheritance, hybrid inheritance, Virtual functions, Friend functions, Generic programming with templates.

Section – C

Streams and Files

Opening and closing a file, File pointers and their manipulations, Sequential Input and output operations, multi-file programs, Random Access, command line argument, string class, Date class, Array class, List class, Queue class, User defined class, Generic Class.

Exception Handling

List of exceptions, catching exception, handling exception,

Section - D

Graphics

Text Mode, Graphics mode functions, Rectangles, and Lines, Polygons & Inheritance, Sound & Motion, Text in Graphics Mode.

Standard Template Library

Standard Template Library, Overview of Standard Template Library, Containers, Algorithms, Iterators, Other STL Elements, Container Classes, General Theory of Operation, Vectors.

Text and Reference Books

1. Robert Lafore, "Object Oriented Programming in Turbo C++", Galgotia Publications,
2. Balagurusamy, 'Object Oriented programming with C++", Tata McGraw Hill.
3. Bjarne Strustrup, "The C++ programming Language", Addison Wesley,
4. Booch, "Object Oriented Analysis and Design with Applications, Addison Wesley.5. Chair H. Pappas & William H. Murray, "The Complete Reference Visual C++", TMH.

Digital Electronics (ECE – 3003)

Course Code	**EC - 3003**	**Credits-3**	**L-3, T-1, P-0**
Name of the Course	**Digital Electronics**		
Lectures to be Delivered	**52 (1 Hr Each) (L=52 for each semester)**		
Semester End Examination	**Max Marks: 100**	**Min Pass Marks: 40**	**Maximum Time: 3 hrs**
Continuous Assessment (based on sessional tests (2) 50%, Tutorials/Assignments 30%, Quiz/Seminar 10%, Attendance 10%)			**Max Marks: 50**

Instructions

1. **For Paper Setters:** The question paper will consist of five sections A, B, C, D & E. Section E will be compulsory, it will consist of a single question with 10-20 subparts of short answer type, which will cover the entire syllabus and will carry 20% of the total marks of the semester end examination for the course. Section A, B, C & D will have two questions from the respective sections of the syllabus and each question will carry 20% of the total marks of the semester end examination for the course.
2. **For candidates:** Candidates are required to attempt five questions in all selecting one question from each of the sections A, B, C & D of the question paper and all the subparts of the questions in Section E. Use of non- programmable calculators is allowed.

Course Description:

Section-A

Number Systems And Boolean Algebra: Subtraction using 1's & 2's complements and using 9's&10's complements, Binary codes, Error detecting and Correcting codes, Theorems of Boolean algebra, Canonical forms, Logic gates.

Section-B

Combinational Circuits: Representation of logic functions, Simplification using Karnaugh map, Tabulation method, Implementation of combinational logic using standard logic gates, Multiplexers and De multiplexers, Encoders and Decoders, Code Converters, Adders, Subtractors, Parity Checker and Magnitude Comparator.

Section-C

Sequential Circuits: Flip flops - SR, JK, D and T flip flops - Level triggering and edge triggering, Excitation tables - Counters - Asynchronous and synchronous type Modulo counters, design with state equation state diagram, Shift registers, type of registers, circuit diagrams

Digital Logic Families: Introduction to bipolar Logic families: RTL, DCTL, DTL, TTL, ECL and MOS Logic families: NMOS, PMOS, CMOS, Details of TTL logic family - Totem pole, open collector outputs, TTL subfamilies, Comparison of different logic families.

Section-D

D/A And A/D Converters: Weighted resistor type D/A Converter, Binary ladder D/A converter, Steady state accuracy test, D/A accuracy and resolution, Parallel A/D Converter, counter type A/D converter, Successive approximation A/D converter, Single and Dual slope A/D converter, A/D accuracy and resolution.

Semiconductor Memories: Memory organization, Classification, and characteristics of memories, Sequential memories, ROMs, R/W memories, Content Addressable memories, Charged-Coupled Device memory, PLA, PAL and Gate Array.

Text Books:

1. Mano M. Morris, "Digital Design", 3rd edition, PHI
2. Jain R. P. "Modern Digital Electronics", 3rd edition, Tata McGraw-Hill 2003.
3. Malvino and Leach "Digital principles and Applications", 5th edition, Tata McGraw Hill, 2003.

4. James W. Bignell and Robert Donovan, “Digital Electronics”, 5th edition, Delmar Publishers, 2007.
5. Flecther “An Engineering Approach to Digital Design”, 1st edition, PHI, 2009.
6. Tocci Ronald J. “Digital Systems-Principles and Applications” 10th edition, PHI, 2009.
7. Flecther “An Engineering Approach to Digital Design”, 1st edition, PHI,2009.
8. Tocci Ronald J. “Digital Systems-Principles and Applications” 10th edition, PHI, 2009.

Principles of Engineering Economics & Management (HS-3001)

Name of the Course	Principles of Engineering Economics & Management		
Course Code	HS – 3001	Credits: 2	L-3, T-1, P-0
Lectures to be delivered	52 (1 Hr Each) (L = 39, T = 13 for each semester)		
Semester End Examination	Max. Time: 3 hrs.	Max. Marks: 100	Min. Pass Marks: 40
Continuous Assessment (based on sessional tests 50%, Tutorials/Assignments 30%, Quiz/Seminar 10%, Attendance 10%)	Max. Marks: 50		

INSTRUCTIONS

For Paper Setters: The question paper will consist of five sections A, B, C, D & E. Section E will be compulsory, it will consist of a single question with 10-20 subparts of short answer type, which will cover the entire syllabus and will carry 20% of the total marks of the semester end examination for the course. Section A, B, C & D will have two questions from the respective sections of the syllabus and each question will carry 20% of the total marks of the semester end examination for the course.

For candidates: Candidates are required to attempt five questions in all selecting one question from each of the sections A, B, C & D of the question paper and all the subparts of the questions in Section E. Use of non-programmable calculators is allowed.

Course Description:

Section A

ECONOMICS: Definitions; Nature & scope of Economics; Economics Systems-meaning of Capitalism; Socialism & mixed economy. DEMAND AND SUPPLIES ANALYSIS: Law of demand and supply, exception to the law of demand; Elasticity of demand and supply and their types; Methods of measuring elasticity of demand and supply.

Section B

THEORY OF PRODUCTION: Scales of production, Law of returns; Break even analysis. MONETARY SYSTEM: Monetary policy Meaning; objectives, methods; Fiscal policy Meaning & objectives of fiscal policy in a developing country like India; Functions of Reserve Bank of India and commercial banks. ECONOMICS & BUSINESS ENVIRONMENT: Privatization; Growth of private capitalism in India; Business/Trade Cycles – Meaning; Characteristics & classification; foreign capital & economic development.

Section C

MANAGEMENT PRINCIPLES: Meaning & types of Management; Concept of Scientific Management; Management By Objectives; System Approach to Management. FINANCIAL MANAGEMENT: Meaning; Functional areas of financial management; Sources of Finance; Meaning of financial accounting; accounting principles-concepts & conventions; Importance of final accounts –profit & loss a/c and balance sheet; Need and importance of capital budgeting. MARKETING MANAGEMENT: Introduction to marketing management; Market segmentation; Developing & managing advertising programs; Deciding on media & measuring effectiveness.

Section D

PRODUCTION MANAGEMENT: Procedure for production planning & Control; Plant Location & Lay-out; Routing; Scheduling; CPM & PERTQUALITY MANAGEMENT: Statistical Quality Control; Introduction Control Charts; X Charts; R Charts; Control Charts for C (N. of defects per unit); Control chart for P (Fraction Defective), Advantages &Limitations of SQC Quality Circles: Structure; Functions & Limitations.

Preferred Books:

1. Business Organisation & Management by B.P.Singh, T.N.Chabra, Dhanpat Rai & Sons

2. Modern Economic Theory by K .K. Dewett, S.Chand & Co.
3. Mark eting Management by Philip Kotler, Prentice Hall of India
4. Financial Management by I.M. Pandey, Vikas Publishing House
5. Indian Economic by Ruddar Dutt, K. P. M. Sundaram, S.Chand & Co
6. Advanced Economic Theory by H.L.Ahuja, S .Chand & Co
7. Production Operation Management by Dr. B.S. Goel, Pragati Prakashan
8. Statistical Quality Control by Grant, Leaven worth, Tata Mc. Graw Hill
9. Personnel Management by, Edwin B.Flippo, Tata Mc. Graw Hll
10. Management – A Global Perespective by Grant, Leaven worth ,TM

Data structures Laboratory (CS-3051)

Course Code	**CS-3051**	**Credits : 1**	**L-0, T-0, P-2**
Name of the Course	**Data Structures Laboratory**		
Lectures to be delivered	**26 hours of Lab sessions**		
Semester End Examination	**Max. Time : 3 hrs**	**Max. Marks : 50**	**Min. Pass Marks : 20**
Continuous Assessment	**(Based on Lab work 30%, Lab record 30%, Viva 30%, Attendance 10%)**	**Max. Marks: 50**	**Min. Pass Marks: 25**

Instructions for paper setter / candidates

Laboratory examination will consist of two parts:

i. Performing a practical exercises assigned by the examiner (25 marks).
ii. Viva-voce examination (25 marks)

Viva-voce examination will be related to the practicals performed / project executed by the candidate related to the paper during the course of the semester.

LIST OF EXPERIMENTS

1. Write a program to insert and delete an element at a specified location in an array.
2. Write a program to print array elements in row and column major order.
3. Write a program to search an element in an array using Linear Search.
4. Write programs to search an element in the array using Binary Search.
5. Write a menu driven program to perform various operations on strings (string length, reverse, concatenate, comparison) using user defined programs.
6. Write a program to implement stack using arrays.
7. Write a program to implement queue using arrays.
8. Write a menu driven program for matrices to do the following operation depending on whether the operation requires one or two matrices
   - Addition and Subtraction of two matrices
   - Finding upper and lower triangular matrices
   - Trace of a matrix, Transpose of a matrix, Check of matrix symmetry
9. Write a program to implement Binary search tree.
10. Write a program to perform insertion & deletion operation on Binary Search trees.
11. Write a program for implementation of a file and performing operations such as insert, delete and update a record in a file.
12. Write a program to create a linked list & display elements of a linked list.
13. Create a linked list and perform the following operation on it
    a) Add a node b) Delete a node c) Count no. of nodes
14. Write a program to implement breadth first search on a graph.
15. Write a program to implement depth first search on a graph.
16. Sorting: Bubble sort, Merge sort, Insertion sort, Selection sort, Radix Sort, Quick Sort

Semester - III

Object Oriented Programming Laboratory with C++ (CS-3052)

Course Code	**CS-3052**	**Credits : 2**	**L-0, T-0, P-2**
Name of the Course	**Object Oriented Programming Laboratory C++**		
Lectures to be delivered	**26 hours of Lab sessions**		
Semester End Examination	**Max. Time : 3 hrs**	**Max. Marks : 50**	**Min. Pass Marks : 20**
Continuous Assessment	**(based on Lab work 30%, Lab record 30%, Viva 30%, Attendance 10%)**	**Max. Marks: 50**	**Min. Pass Marks: 25**

Instructions for paper setter / candidates

Laboratory examination will consist of two parts:

(i) Performing a practical exercises assigned by the examiner (25 marks).

(ii) Viva-voce examination (25 marks)

Viva-voce examination will be related to the practicals performed / project executed by the candidate related to the paper during the course of the semester.

LIST OF EXPERIMENTS

1. Raising a number n to a power of p is the same as multiplying n by itself p times. Write a function called power( ) that takes a double value for an int value for p and returns the result as double value . Use a default argument of 2 of p, so that if this argument is omitted, the number will be squared. Write a main ( ) function that gets values from the user to test this function .
2. A point on the two dimensional plane can be represented by two numbers: an X coordinate and a Y coordinate. For example (4,5) represents point 4 unit to the right of origin along the X axis and 5 units up the y-axis . The sum of the two points can be defined as new point whose X and Y coordinates.

   Write a program that uses a structure called point to model a point . Define three points and have the user input values to two of them. Than set the third point equal to the sum of the other two. And display the value of new points. Interaction with the program might look like this.

   Enter Coordinate of P1: 3 4
   Enter Coordinate of P2: 5 7
   Coordinates of P1+P2 are : 8 11

3. Create the equivalent of four function calculator . The program should request the to user to enter a number , an operator and another number . It should carry out the specified arithmetical operation: adding, subtracting, multiplying ,or dividing the two numbers. (it should use a switch statement to select the operation) finally it should be display the result. When it finishes the calculation, the program should ask if the user want to do another calculation. The response can be 'Y' or 'N', Some sample interaction with the program might look like this.

   Enter first number ,operators and second
   number 12+100 Answer =112
   Do another (Y/N)?N

4. A phone no. such as (212)767-8900 , can be thought of as having three parts area code(212), the exchange (767) and the number (8900). Write a program that uses a structure to store these three parts of phone both no. separately . Call the structure phone . create two structure Enter your area code Exchange and number : 415 555 1212
   My number is (415)555-1212
5. Create two classes DM and DB which stores the value of distances DM stores distance in meters and centimeters and DB in feet and inches . Write a program that can read value for the classes objects and add one object of DM with another object DB.
   Use a friend function to carry out the addition operation .The object that stores the result may be a Dm object or DB object depending on the units in which result are required .
   The display should be in the format of feet and inches or meters and centimeters depending on the object on display.
6. Create a class rational which represents numerical value by two double value NUMERATOR & DENOMENATOR . Include the following public member functions:
   - Constructor with no arguments.(defaults)
   - Constructor with two arguments.
   - Void reduce() that reduce the rational number by eliminating the highest common factor between the numerator and denominator .
   - Overload +operator to add two rational number
   - Overload operator >> operator to be enabled input through cin
   - Overload <<operator to be enabled input through count.

   Write a main ( ) to test all the functions in the class
7. Consider the following class definition class father {
   Protected : int age;
   Public:
   Father (int x) {age = x;} Virtual void iam ()
   {
   {cout <<"I AM THE FATHER , my age is ",<<age<<endl;}
   };
   Derive the two classes son and daughter from the above classes and for each define iam() to write our similar but appropriate message .You should also define suitable constructors for these classes Now write a main ( ) that creates objects of three classes and then call iam( ) them .Declare pointer to father , successively assign addresses of object of the two derived classes to this pointer and in each case , call iam( ) through the pointer to demonstrate polymorphism in action.
8. Write a program that create a binary files by reading the data from the students from the terminal. The data of each student consist of roll no, name(a string of 30 or lesser no. of character) and marks.
9. A hospital wants to create a database regarding its indoor patients. The information to store include.
   a) Name of the patient
   b) Date of admission
   c) Disease
   d) Date of discharge

   Create a structure to store the data (year, month, date as its members). Create a base class to store the above information. The member function should include function to enter information and display a list of all the patients in database Create a drive class to store the age of patients. List the information about all to store the age of the patients. List the information about all the pediatric (less then twelve years in age)

10. Makes a class Employee with the name and salary . Makes a class manager inherit from the Employee Add an instance variable named :department, type : string. Supply a method to String that print the manager's name, department and salary. Make a class Executive inherit from information store in the manager super class object . Supply a test program that test these classes and methods.
11. Imagine a tollbooth with a class called Toll booth . The two data item are a type unsigned into to hold the total number of cars and type double to hold the total amount of money collected . A constructor initializes both these to 0. A member function called nopaycar( ). Increments the car total and adds 0.50 to the cash total. Another function, called nopaycar( ), increment the car total but adds nothing to the cash total. Finally , a member function called display the two totals . Include a program to test this class . This program should allow the user to push one key to count paying a car and another to count a non paying car . Pushing the ESC key should cause the program to print out the total cars and total cash and then exit.
12. Write a function called reverse it ( ) that reverses a string(an array of char) use a for loop that swap the first and last characters, then the second and next to last character and so on .

    the string should be passed to reversesit ( ), and print out the result. Use an input method that allows embedded blanks. Test the program with Napoleon's famous phrase , "Able was ere I saw Elba".
13. Create some objects of the string class, and put them in a Deque – some at the head of the Deque and some at the tail. Display the contents of the Deque using the for Each( ) function and a user written display function . Then search the Deque for a particular strings, using the first That ( ) Function and display any string that match, finally remove all the item from the date using the get left( ) Function and display each item. Not ice is the order in which the item are displayed: Using Get Left ( ), Those inserted on the left (head),of the Deque are removed in "last and first out" order while those put on the right side are removed in "first in first out" order. The opposite would be true if Get right ( ) were used
14. Assume that a blank maintain two kinds of accounts for customer. One called as saving accounts and another is current account . The saving account provides compound interest and withdrawal facility but no cheque book facility, The current account provides cheque book facility but no interest Current account holders should also maintain a minimum balance and if the balance false below this level , a service charge is imposed.

    Create a class account that store customer name, account number and type of account. From this drive the classes cur_acct and sav_account to make them more specific to their requirement. Include necessary member function in order to achieve the following task

    a) Accept deposit from a customer and update the balance
    b) Display the balance
    c) Compute and deposit interest
    d) Permit withdrawal and update the balance
    e) Check for the minimum balance, impose penalty ,necessary and update the balance. f) Do not use any

    constructor , use member function to initialize the class members
15. Create a base class called shape .Use this class to store two double type values that could be used to compute the area of figure, Derive to specific classes called triangle and rectangle from the base shape . Add to the base class, a member function get data ( ) to initialize base class data member and another member function display area ( ), To compute and display the area of figures make display area ( ) as virtual function and redefine this function in the derived classes to suit the requirements.

Using this three classes design a program that will accept dimension of triangle or rectangle interactively and display the area

Remember the two value given as input will be treated as length of two sides in the case of rectangle and as base and height in the case of triangle and used as follows

Area of rectangle= x * y

Area of triangle =1/2 * x * y

Programming of exercise in C++ in the form of project (based on "object oriented programming in TURBO C++") , Robert lafore , Galgotia Publication Pvt. Ltd.1994 to be done in consultation with the faculty incharge for the course

Note: Record to be maintained both electronically and hard copy of evaluation

FOURTH SEMESTER

Semester - IV
Numerical Methods (ES - 4001)

Name of the Course	**Numerical Methods**		
Course Code	**ES – 4001**	**Credits: 3**	**L-3, T-1, P-0**
Lectures to be delivered	**52 (1 Hr Each) (L = 39, T = 13 for each semester)**		
Semester End Examination	**Max. Time: 3 hrs.**	**Max. Marks: 100**	**Min. Pass Marks: 40**
Continuous Assessment (based on sessional tests 50%,Tutorials/Assignments 30%, Quiz/Seminar 10%, Attendance 10%)	**Max. Marks: 50**		

INSTRUCTIONS

1. **For Paper Setters:** The question paper will consist of five sections A, B, C, D & E. Section E will be compulsory, it will consist of a single question with 10-20 subparts of short answer type, which will cover the entire syllabus and will carry 20% of the total marks of the semester end examination for the course. Section A, B, C & D will have two questions from the respective sections of the syllabus and each question will carry 20% of the total marks of the semester end examination for the course.
2. **For candidates:** Candidates are required to attempt five questions in all selecting one question from each of the sections A, B, C & D of the question paper and all the subparts of the questions in Section E.

Use of non- programmable calculators is allowed.

Section-A

Solution of algebraic and transcendental equations: Bisection method, method of false position, secant method, Iteration method Newton-Raphson method.

Solution Of Simultaneous Algebraic Equations : Gaus elimination method, Jacobi's method, Gauss-Seidal method.

Section-B

Finite Differences & Interpolation: Forward and Backward difference operators, Newton's Forward and Backward interpolation formulae, Central Difference Interpolation formulae, Gauss's forward and Backward Interpolation formulae, Lagrange's interpolation formulae and Newton's Divided Difference formulae.

Section- C

Numerical Methods To Solve Differential Equations: Solution of first order differential equations using Taylor's Series, Euler's, Picard's and Runge-Kutta method upto 4^{th} order, Predictor- Corrector methods, Simultaneous differential equations of first order, differential equations of second order.

Section- D

Numerical Integration: Numerical integration using Trapezoidal rule, Simpson's 1/3rd and 3/8th rules, Two point and three point Gauss quadrature method.

Text Books:

1. Sastry SS, Introductary Methods of Numerical Analysis, Prentice Hall of India
2. Grewal, BS, "Numerical Methods", Khanna Publishers
1. Chapra SC and Canale RP, Numerical Methods for Engineers, McGraw Hill Book Company
2. Computer Oriented Numerical Methods By: V. Rajaraman, PHI Learning Pvt. Ltd

Semester - IV
Operating Systems (CS - 4001)

Course Code	**CS - 4001**	**Credits-3**	**L-3, T-1, P-0**
Name of the Course	**Operating Systems**		
Lectures to be Delivered	**52 (1 Hr Each) (L=39, T=13 for each semester)**		
Semester End Examination	**Max Marks: 100**	**Min Pass Marks: 40**	**Maximum Time: 3 hrs**
Continuous Assessment (based on sessional tests (2) 50%, Tutorials/Assignments 30%, Quiz/Seminar 10%, Attendance 10%)			**Max Marks: 50**

Instructions

1. **For Paper Setters:** The question paper will consist of five sections A, B, C, D & E. Section E will be compulsory, it will consist of a single question with 10-20 subparts of short answer type, which will cover the entire syllabus and will carry 20% of the total marks of the semester end examination for the course. Section A, B, C & D will have two questions from the respective sections of the syllabus and each question will carry 20% of the total marks of the semester end examination for the course.
2. **For candidates:** Candidates are required to attempt five questions in all selecting one question from each of the sections A, B, C & D of the question paper and all the subparts of the questions in Section E. Use of non-programmable calculators is allowed.

Course Description:

Section A

What is an Operation System? Simple Batch Systems; Multiprogrammed Batched Systems;Time-Sharing System; Personal-Computer systems; Parallel System; Distributed System; Real- Time Operating Systems. System Components System Calls, System Programs; System Structure; Virtual Machines.

Process concept: Process Scheduling; Operation on processes, Interprocess Communication,CPU Scheduling fundamental concepts, Scheduling criteria; Scheduling Algorithms; Multi- processor Scheduling; Real Time Scheduling, Threads: Overview; Multithreading

Section B

Process Synchronization: Race conditions; mutual exclusion requirements; Critical Section problem; s/w and h/w solutions; Semaphores; monitors; Classical IPC problem and solutions.

Deadlock: System Model; Deadlock Characterization, Methods of Handling Deadlock, deadlock Prevention; Deadlock Avoidance; Deadlock Detection, Recovery from deadlock; Combined approach to deadlock handling

File System Interface: File Concept; Access Methods; Directory Structure; Protection; Consistency Semantics;

File System Implementation: File System Structure; Allocation Methods, Free Space Management Directory Implementation; Efficiency and

Performance; Recovery.

Section C

Memory Management: Logical Versus Physical Address Space, Swapping, Contiguous Allocation; Paging; Segmentation; Segmentation with paging.

Virtual Memory: Demand Paging Performance of Demand Paging page Replacement Page Replacement Algorithms; Allocation of Frames Thrashing; Demand Segmentation; Cache memory and implementation.

Secondary Storage Structure: Disk Structure; Disk Scheduling; Disk Management; Swap-space management; Disk Reliability; Stable-Storage Implementation.

Section D

I/O Systems: I/O hardware; I/O channels; Structure of I/O System; Principles of I/O Software Goals; interrupt handlers; device drivers; device independent I/O software;

Protection: Goals of protection; Domain of protection; Access matrix and its implementation; Revocation of Access Right; Capability- Based Systems; Language Based Protection.

Security: The Security Problem; Authentication; One Time passwords program Threats, System Threats; Threat Monitoring; Encryption and decryption; Computer-Security Classification; An example Security Model: windows NT

Books:

1. Operating System Concepts by Silberschatz & Galvin, Wiley Publication
2. Operating System (5th) – Internals & Design Principles by William Stallings,Prentice Hall
3. Operating Systems by Achiest S. God bole, McGraw Hill Publication
4. Operating Systems by D. M. Dhamdhare, Tmh
5. Understanding Operating System by Flynn & Métiers Thomsan
6. Operating Systems Design & Implementation by Andrew Dagenham, Albert S.Wood Hull, Pearson Publication

Semester - IV
Software Engineering (CS - 4002)

Course Code	**CS - 4002**	**Credits-3**	**L-3, T-0, P-0**
Name of the Course	**Software Engineering**		
Lectures to be Delivered	**52 (1 Hr Each) (L=52 for each semester)**		
Semester End Examination	**Max Marks: 100**	**Min Pass Marks: 40**	**Maximum Time: 3 hrs**
Continuous Assessment (based on sessional tests (2) 50%, Tutorials/Assignments 30%, Quiz/Seminar 10%, Attendance 10%)			**Max Marks: 50**

Instruct
i ons

1. **For Paper Setters:** The question paper will consist of five sections A, B, C, D & E. Section E will be compulsory, it will consist of a single question with 10-20 subparts of short answer type, which will cover the entire syllabus and will carry 20% of the total marks of the semester end examination for the course. Section A, B, C & D will have two questions from the respective sections of the syllabus and each question will carry 20% of the total marks of the semester end examination for the course.
2. **For candidates:** Candidates are required to attempt five questions in all selecting one question from each of the sections A, B, C & D of the question paper and all the subparts of the questions in Section E. Use of non-programmable calculators is allowed.

Course Description:

SECTION-A

Introduction: Need for software engineering, issue in the design of large software, software life cycle models, overview of software development process.

SECTION-B

Software Requirement Analysis and Specification- Requirements Engineering, Crucial process step, State of the practice, problem analysis, Data dictionaries, Entity relationship diagram, code object diagram, approaches to problem analysis, Structured requirements definition, structured analysis & design techniques, Software prototyping, Software requirements specification, Nature of SRS, characteristics of good SRS. Organization of the SRS, Specifying behavioral requirements, finite state machines, decision tables & tree, PDL

SECTION-C

Software Matrics: What and why: Definition, areas of applications, problems during implementation, size matrics, The basic information Flow Model, the more sophisticated information Flow Model, Metrics analysis using statistics for

Assessment, problems with metric data, The common of pool of data. A pattern for successful applications

SECTION-D

Software Project Planning: Cost estimation: Models , Static ,single variable model, Static multivariable model, The constructive cost model: Basic model, International model, Detailed COCOMO Model, The Putnam resource allocation model: The trade off- -of-time versus cost, development sub cycle,

Software Risk Management : what is Risk, typical software risks , Risk management Activities, Risk identification, Risk projection, Risk management activity

Books:

1. Software Engineering- A practitioner's Approach, Roger S. Pressmen, McGraw Hill Publication New Delhi
2. Software Engineering-K.K. Aggarwal & Yogesh, New Age International Publishers
3. Software Engineering- A Systematic Approach by J.S. Dilawari, Paragon Publishers New Delhi

Semester - IV
Analysis & Design of Algorithms (CS - 4003)

Course Code	**CS - 4003**	**Credits-3**	**L-3, T-1, P-0**
Name of the Course	**Analysis & Design of Algorithms**		
Lectures to be Delivered	**52 (1 Hr Each) (L=39, T=13 for each semester)**		
Semester End Examination	**Max Marks: 100**	**Min Pass Marks: 40**	**Maximum Time: 3 hrs**
Continuous Assessment (based on sessional tests (2) 50%, Tutorials/Assignments 30%, Quiz/Seminar 10%, Attendance 10%)			**Max Marks: 50**

Instructions

1. **For Paper Setters:** The question paper will consist of five sections A, B, C, D & E. Section E will be compulsory, it will consist of a single question with 10-20 subparts of short answer type, which will cover the entire syllabus and will carry 20% of the total marks of the semester end examination for the course. Section A, B, C & D will have two questions from the respective sections of the syllabus and each question will carry 20% of the total marks of the semester end examination for the course.
2. **For candidates:** Candidates are required to attempt five questions in all selecting one question from each of the sections A, B, C & D of the question paper and all the subparts of the questions in Section E. Use of non-programmable calculators is allowed.

Course Description:

Section A

Introductory Concepts: The notation of algorithm; fundamentals of algorithmic problem solving; analyzing algorithms; Review of fundamental data structures; Arrays, Stacks; Queue; Linked list Fundamentals of analysis of algorithms efficiency: Asymptotic notation and standard efficiency classes; mathematical analysis of recursive and non-recursive algorithms

Divide and Conquer: General Method; Merge sort; Quick sort; Selection sort; Sorting in Linear time: Counting sort; Radix sort and Bucket sort Search: Linear Search; Binary Search

Section B

Graphs: Review of Graphs; Representation of Graphs; Breadth-first search; Depth-first Search; Topological Sort; Strongly connected Components

Trees: Review of Trees; Minimum spanning tree; Kruskal and Prim's algorithm; Single source shortest paths; Bellaman-Ford algorithm; Single source shortest path in directed acyclic graphs; Dijkstra's algorithm; All pairs shortest paths; Shortest paths and matrix multiplication; Floyd-Warshall algorithm; Johnson's algorithm

Section C

Dynamic Programming: Introduction; Elements of Dynamic Programming; Matrix Chain Multiplication; Longest Common Subsequence; Optimal binary search tree; Knapsack problem; Travelling sales person problem.

Greedy Method: An activity selection problem; Elements of Greedy Programming; Huffmann codes; A task scheduling problem

Backtracking and Branch and Bound: The 8 Queens problem; Graph coloring; Hamiltonian cycles; Least Cost Search(LC); The 15 puzzle

Bounding: Fifo branch and bound; LC branch and bound.

Section D:

Maximum Flow: Flow Networks; The Ford-Fulkerson method; Maximum Bipartite matching; Sorting Networks: Comparison networks; Zero-one principle; Bitonic sorting network; merging network; sorting network

NP hard and NP complete problems: P; NP; NP hard and NP complete problems; Cook's theorem(proof not required); Basic introduction to clique problem; vertex cover problem; Hamiltonian cycle problem; Approximation algorithms; vertex cover problem; Travelling salesman problem.

Text and Reference Books

1. Cormen, Leiserson, Rivest, Stein "Introduction to Algorithms" The MIT Press &McGrawHill Publication
2. Horowitz Ellis And Sartaj Sahni "Fundamentals of Computer Algorithms" Universities Press
3. Anany V. Levitin "Introduction to Design and analysis of algorithms" Pearson Education Publisher
4. Aho-Hopcroft and Ullman "The Design and Analysis of computer algorithms" Addison-Wesley Publishing Company

Theory of Computation (CS-4004)

Course Code	**CS - 4004**	**Credits-3**	**L-3, T-0, P-0**
Name of the Course	**Theory of Computation**		
Lectures to be Delivered	**52 (1 Hr Each) (L=52 for each semester)**		
Semester End Examination	**Max Marks: 100**	**Min Pass Marks: 40**	**Maximum Time: 3 hrs**
Continuous Assessment (based on sessional test (2) 50%, Tutorials/Assignments 30%, Quiz/Seminar 10%, Attendance 10%)			**Max Marks: 50**

Instructions

1. **For Paper Setters:** The question paper will consist of five sections A, B, C, D & E. Section E will be compulsory, it will consist of a single question with 10-20 subparts of short answer type, which will cover the entire syllabus and will carry 20% of the total marks of the semester end examination for the course. Section A, B, C & D will have two questions from the respective sections of the syllabus and each question will carry 20% of the total marks of the semester end examination for the course.
2. **For candidates:** Candidates are required to attempt five questions in all selecting one question from each of the sections A, B, C & D of the question paper and all the subparts of the questions in Section E. Use of non-programmable calculators is allowed.

Course Description:

Section A

Finite Automata and Regular Expression: Finite State System; Basic Definitions; Non-Deterministic finite Automata (NDFA); Deterministic finite Automata(DFA);Equivalence of DFA and NDFA Finite Automata with E- moves; Regular expression; Equivalence of finite Automata and expression; Regular expression conversion and vice–versa.

Section B

Introduction to Machines: Concept of basic machines; Properties and limitation of FSM; Moore and Mealy Machines; Equivalence of Moore and Mealy Machines; Conversion of NFA to DFA by Arden's method

Properties of Regular Sets: The Pumping Lemma for Regular sets; Application of the pumping lemma; Closure properties of regular sets; Myhill-Nerode Theorem and minimization of Finite Automata; Minimization Algorithm; Kleene's Theorem.

Section C

Grammars: Definition; Context Free and context sensitive grammar; Ambiguity; Regular grammar; Reduced forms; Removal of useless Symbols and unit production; Chomsky Normal Form(CNF); Griebach Normal Form(GNF).

Pushdown Automata: Introduction to push-down machines; Application of pushdown machines.

Section D

Turing Machines: Deterministic and Non-Deterministic Turing Machines; Design of T.M; Halting problem of T.M; PCP problem.

Chomsky Hierarchy: Chomsky hierarchies of grammars; Unrestricted grammar; Context sensitive Language; Relation between language of classes.

Computability: Basic Concepts; Primitive Recursive Functions.

Text and Reference Books

1. Hopcroaft & O.D.Ullman, R.Motwani: Introduction to Automata Theory, languages & computations, Pearson Publication
2. K.L.P.Mishra & N.Chandershekaran: Theory of Computer Sc. (Automata, Language & Computation), Prentice-Hall of India
3. Peter Linz: Introduction to formal language & Automata, Jones and Bartlett Publishers
4. John C. Martin: Introduction to Languages and the Theory of Computation, McGraw Hill Publication.
5. Introduction to the Theory of Computation by Michael Sipser, PWS Publishing company
6. Computation Complexity -A Modern Approach by Sanjeev Arora and Boaz Barak

Python Programming (CS-4005)

Course Code	CS-4005	Credits-3	L-3, T-1, P-0
Name of the Course	Python Programming		
Lectures to be Delivered	52 (1 Hr Each) (L=39, T=13 for each semester)		
Semester End Examination	Max Marks: 100	Min Pass Marks: 40	Maximum Time: 3 hrs
Continuous Assessment (based on sessional tests (2) 50%, Tutorials/Assignments 30%, Quiz/Seminar 10%, Attendance 10%)			Max Marks: 50

Instructions

For Paper Setters: The question paper will consist of five sections A, B, C, D and E. Section E will be Compulsory, it will consist of a single question with 10-20 subparts of short answer type, which will cover the entire syllabus and will carry 20% of the total marks of the semester end examination for the course. Section A, B, C and D will have two questions from the respective sections of the syllabus and each question will carry 20% of the total marks of the semester end examination for the course.

For Candidates: Candidates are required to attempt five question in all selecting one question from each of the section A, B, C and D of the question paper and all the subparts of the questions in section E. Use of non-programmable calculators is allowed.

Section A

Parts of Python Programming Language, Identifiers, Keywords, Statements Expressions, Variables, Operators, Precedence and Associativity, Data Types, Indentation, Comments, Reading Input, Print Output, Type Conversions, The type() Function and Is Operator, Dynamic and Strongly Typed Language, Control Flow Statements, The if Decision Control Flow Statement, The if…else Decision Control Flow Statement, The if…elif…else Decision Control Statement, Nested if Statement, The while Loop, The for Loop, The continue and break Statements, Catching Exceptions Using try and except Statement, Functions, Built-In Functions, Commonly Used Modules, Function Definition and Calling the Function, The return Statement and void Function, Scope and Lifetime of Variables, Default Parameters, Keyword Arguments, *args and **kwargs, Command Line Arguments.

Section B

Strings**,** Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings, **Lists,** Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions Used on Lists, List Methods, The del Statement.

Dictionaries, Creating Dictionary, Accessing and Modifying key:value Pairs in Dictionaries, Built-In Functions Used on Dictionaries, Dictionary Methods, The del Statement, **Tuples and Sets,** Creating Tuples, Basic Tuple Operations, Indexing and Slicing in Tuples, Built-In Functions Used on Tuples, Relation between Tuples and Lists, Relation between Tuples and Dictionaries, Tuple Methods, Using zip() Function, Sets, Set Methods, Traversing of Sets, Frozenset.

Section C

Files, Types of Files, Creating and Reading Text Data, File Methods to Read and Write Data, Reading and Writing Binary Files, The Pickle Module, Reading and Writing CSV Files, Python os and os.path Modules, **Regular Expression Operations,** Using Special Characters, Regular Expression Methods, Named Groups in Python Regular Expressions, Regular Expression with glob Module.

Section D

Object-Oriented Programming, Classes and Objects, Creating Classes in Python, Creating Objects in Python, The Constructor Method, Classes with Multiple Objects, Class Attributes versus Data Attributes, Encapsulation, Inheritance, The Polymorphism

Books:

TEXT BOOK

1. Gowrishankar S, Veena A, "Introduction to Python Programming", 1st Edition, CRC Press/Taylor & Francis, 2018. ISBN-13: 978-0815394372

REFERENCE BOOKS / WEBLINKS:

1. Jake VanderPlas, **"**Python Data Science Handbook: Essential Tools for Working with Data"**,** 1st Edition, O'Reilly Media, 2016. ISBN-13: 978-1491912058
2. Aurelien Geron, Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems", 1st Edition,O'Reilly Media, 2017. ISBN – 13: 978-1491962299.
3. Wesley J Chun, "Core Python Applications Programming"**,** 3rd Edition, Pearson Education India, 2015. ISBN-13: 978-9332555365
4. Miguel Grinberg, "Flask Web Development: Developing Web Applications with Python", 2nd Edition, O'Reilly Media, 2018. ISBN-13: 978-1491991732.

Operating System Laboratory (CS-4051)

Course Code	CS-4051	Credits-1	L-0, T-0, P-2
Name of the Course	Operating System Laboratory		
Lectures to be Delivered	26 Hrs. of Lab work (2hrs. each per week)		
Semester End Examination	Max Marks: 50	Min Pass Marks: 20	Maximum Time: 3 hrs
Continuous Assessment	Based on Lab work 30%, Lab Record 25%, Viva 25% & Attendance 20%	Max Marks: 50	Min Pass Marks: 25

Instructions for paper setter / Candidates

Laboratory examination will consist of two parts:

1. Performing a practical examination assigned by the examiner. (25 marks)
2. Viva-voce examination. (25 marks)

Viva-voce examination will be related to the practical performed/projects executed by the candidate related to the paper during the course of the semester.

CASE STUDIES on the following operating system to be done in consultation with the faculty incharge for the course:

1. Singal User System :MS-DOS and Windows98

2. Network Operating System: Windows 2000/Windows NT

3. Multiuser System :Unix/Linux
4. Study the Linux operating system and implement various commands and shell scripting.
5. Implement the process synchronization using semaphores.
6. Write the program to mount the various devices (i.e. floppy, CD-Rom etc)
7. Write a program do the following thing…
    a. Find the attribute of file.
    b. To change the attribute of file.
    c. Create the directory.
    d. Delete the directory.
    e. Create the file.
    f. Delete the file.

Analysis & Design of Algorithms Laboratory (CS – 4052)

Course Code	CS - 4052	Credits : 1	L-0, T-0, P-2
Name of the Course	Analysis & Design of Algorithms Laboratory		
Lectures to be delivered	26 hours of Lab sessions		
Semester End Examination	Max. Time : 3 hrs	Max. Marks : 50	Min. Pass Marks : 20
Continuous Assessment	Based on Lab work 30%, Lab record 30%, Viva 30%, Attendance 10%	Max. Marks: 50	Min. Pass Marks: 25

Instructions for paper setter / candidates

Laboratory examination will consist of two parts:

(i) Performing a practical exercises assigned by the examiner (25 marks).
(ii) Viva-voce examination (25 marks)

Viva-voce examination will be related to the practicals performed / project executed by the candidate related to the paper during the course of the semester

List of Experiments:

1) a) Obtain the Topological ordering of vertices in a given digraph.
   b) Compute the transitive closure of a given directed graph using Warshall's algorithm.

2) Implement 0/1 Knapsack problem using Dynamic Programming.
3) From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijikstra's algorithm.

4) Find Minimum Cost Spanning Tree of a given undirected graph using Kristal's algorithm.
5)
   a. Print all the nodes reachable from a given starting node in a digraph using BFS method.
   b. Check whether a given graph is connected or not using DFS method.

6) Find a subset of a given set S = {sl,s2,.....,sn} of n positive integers whose sum is equal to a given positive integer d. For example, if S= {1, 2, 5, 6, 8} and d = 9 there are two solutions{1,2,6} and {1,8}.A suitable message is to be displayed if the given problem instance doesn't have a solution.
7) Implement any scheme to find the optimal solution for the Traveling Salesperson problem and then solve the same problem instance using any approximation algorithm and determine the error in the approximation.
8) Find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm.
9) Implement All - Pairs Shortest Paths Problem using Floyd's algorithm. Parallelize this algorithm, implement it using Open and determine the speed-up achieved.
10) Implement N Queen's problem using Back Tracking.

Online Resource:

http://vtucsenotes.files.wordpress.com/2013/06/design-and-analysis-of-algorithms-laboratory1.pdf

Python Programming Laboratory (CS-4053)

Course Code	CS - 4053	Credits : 1	L-0, T-0, P-2
Name of the Course	Python Programming Laboratory		
Lectures to be delivered	26 hours of Lab sessions		
Semester End Examination	Max. Time : 3 hrs	Max. Marks : 50	Min. Pass Marks : 20
Continuous Assessment	Based on Lab work 30%, Lab record 30%, Viva 30%, Attendance 10%	Max. Marks: 50	Min. Pass Marks: 25

Instructions

Laboratory examination will consist of two parts:

1. Performing a practical examination assigned by the examiner. (25 marks)
2. Viva-voce examination. (25 marks)

Viva-voce examination will be related to the practical performed/projects executed by the candidate related to the paper during the course of the semester

Programs Implementation Using Python Programming Language:

1.Write Program to print "HELLO PYTHON".
2. Write a Program to get input from user and print it on screen.
3. Write a program to swap two numbers.
4. Write a Program to perform basic calculator operations.
5. Write a Program to check if string is palindrome or not.
6. Write a Program to illustrate common string operations in python.
7. Write a Program to print prime numbers.
8. Write a Program that uses ten different inbuilt Mathematical functions.
9. Write a Program to find factorial of given number.
10. Write a Program to reverse the string.
11. Write a Program to print Fibonacci series.
12. Write a Program to explain different types of loop control statements.
13. Write a Program to show different types of functions in Python.
14. Write a Program showing concept of 'Scope of Variable'.
15. Write a Program to show use of five dictionary functions.
16. Write a Program to show types of inheritance in Python.
17. Write a Program to explain method overloading and method overriding.
18. Write a Program to show Exception Handling in Python.
19. Write a program to explain User-Defined Exception.
20. Write a Program to short the list entered by the user.
21. Write a Program to delete and update the element in list.
22. Write a Program the shows the use of mkdir(), chdir(), getcwd(), rmdir() function.
23. Write a Program to write "Hello Python" in file.
24. Write a Program to explain match and search functions. (Related to Regular Expressions)
25. Write a Program that works as chat application between client and server.
26. Write a Program to get following output using GUI.

Software Engineering Project (CS-4054)

Course Code	CS-4054	Credits : 1	L-0, T-0, P-2
Name of the Course	Software Engineering Project		
Lectures to be delivered	26 hours of Lab sessions		
Semester End Examination	Max. Time : 3 hrs	Max. Marks : 50	Min. Pass Marks : 20
Continuous Assessment	(based on Lab work 30%,Lab record 30%,Viva 30%, Attendance 10%)	Max. Marks: 50	Min. Pass Marks: 25

Aim of this Project

Aim of this Project is to equip students in the methodology of Software Engineering of a Live Project in the institute in which he is studying or in a place of work such as Bank, School, College and office in the vicinity of the institute. This will be a guide Project under the Close supervision of the faculty of the institute. Project should be presented in the form of a project report giving a candidate system for solving a life problem.