ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES
(Autonomous)
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
B TECH COURSE PROPOSED SCHEME UNDER AUTONOMOUS SYSTEM 2015-16

1st Year Semester I

CODE NO	SUBJECT NAME	Instruction periods per week				Max Marks		Credits
		Lecture	Tutorial	Lab	Total	Sessional Marks	Semester End Marks	
CSE111	ENGLISH	3	1	-	4	40	60	3
CSE112	ENGINEERING MATHEMATICS-I	3	1	-	4	40	60	3
CSE113	ENGINEERING CHEMISTRY	3	1	-	4	40	60	3
CSE114	PROFESSIONAL ETHICS &HUMAN VALUES	2	1	-	3	100	-	2
CSE115	BASIC ELECTRONICS ENGINEERING	3	1	-	4	40	60	3
CSE116	ENGINEERING CHEMISTRY LAB	-	-	3	3	50	50	2
CSE117	PROGRAMMING WITH C-LAB	-	1	3	4	50	50	3
CSEAC1	NCC/NSS/SPORTS	-	-	3	3	-	-	0
Total		***14***	***6***	***9***	***29***	***360***	***340***	***19***

1st Year Semester II

CODE NO	SUBJECT NAME	Instruction periods per week				Max Marks		Credits
		Lecture	Tutorial	Lab	Total	Sessional Marks	Semester End Marks	
CSE121	ENGNEERING MATHEMATICS-II	3	1	-	4	40	60	3
CSE122	ENGINEERING PHYSICS	3	1	-	4	40	60	3
CSE123	ENVIRONMENTAL SCIENCES	3	1	-	4	40	60	3
CSE124	ENGINEERING DRAWING	1	-	3	4	40	60	3
CSE125	ELEMENTS OF ELECTRICAL ENGINEERING	3	1	-	4	40	60	3
CSE126	ENGINEERING PHYSICS LAB	-	-	3	3	50	50	2
CSE127	LANGUAGE LAB	-	-	3	3	50	50	2
CSE128	OBJECT ORIENTED PROGRAMMING WITH C++ --LAB	-	1	3	4	50	50	3
CSE129	WORKSHOP	-	-	3	3	50	50	2
CSEAC2	NCC/NSS/SPORTS	-	-	3	3	-	-	0
Total		***13***	***5***	***18***	***36***	***400***	***500***	***24***

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES
(Autonomous)
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
B TECH COURSE PROPOSED SCHEME UNDER AUTONOMOUS SYSTEM 2015-16

2nd Year Semester I

CODE NO	SUBJECT NAME	Instruction periods per week				Max Marks		Credits
		Lecture	Tutorial	Lab	Total	Sessional Marks	Semester End Marks	
CSE211	DATA STRUCTURES & ALGORITHMS	4	1	-	5	40	60	4
CSE212	DIGTAL LOGIC DESIGN	3	1	-	4	40	60	3
CSE213	DISCRETE MATHEMATICAL STRUCTURES	4	1	-	5	40	60	4
CSE214	OBJECT ORIENTED PROGRAMMING WITH JAVA	3	1	-	4	40	60	3
CSE215	PROBABILITY, STATISTICS & QUEUING THEORY	4	1	-	5	40	60	4
CSE216	DATA STRUCTURES LAB	-	-	3	3	50	50	2
CSE217	DIGITAL ELECTRONICS LAB	-	-	3	3	50	50	2
CSE218	JAVA LAB	-	-	3	3	50	50	2
Total		***18***	***5***	***9***	***32***	***350***	***450***	***24***

2nd Year Semester II

CODE NO	SUBJECT NAME	Instruction periods per week				Max Marks		Credits
		Lecture	Tutorial	Lab	Total	Sessional Marks	Semester End Marks	
CSE221	DATA COMMUNICATIONS	4	1	-	5	40	60	4
CSE222	MICROPROCESSORS AND INTERFACING	3	1	-	4	40	60	3
CSE223	OPERATING SYSTEMS	4	1	-	5	40	60	4
CSE224	COMPUTER ORGANIZATION	4	1	-	5	40	60	4
CSE225	FORMAL LANGUAGES AND AUTOMATA THEORY	4	1	-	5	40	60	4
CSE226	MICROPROCESSOR & INTERFACING LAB	-	-	3	3	50	50	2
CSE227	OPERATING SYSTEMS LAB	-	-	3	3	50	50	2
CSE228	HARDWARE LAB	-	-	3	3	50	50	2
Total		***19***	***5***	***9***	***33***	***350***	***450***	***25***

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES
(Autonomous)
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
B TECH COURSE PROPOSED SCHEME UNDER AUTONOMOUS SYSTEM 2015-16
3rd Year Semester I (Tentative)

CODE NO	SUBJECT NAME	Instruction periods per week				Max Marks		Credits
		Lecture	Tutorial	Lab	Total	Sessional Marks	Semester End Marks	
CSE311	COMPUTER NETWORKS	4	1	-	5	40	60	4
CSE312	DATA BASE MANAGEMENT SYSTEMS	4	1	-	5	40	60	4
CSE313	COMPUTER GRAPHICS	4	1	-	5	40	60	4
CSE314	OPEN ELECTIVE I	3	1	-	4	40	60	3
CSE315	DESIGN & ANALYSIS OF ALGORITHMS	4	1	-	5	40	60	4
CSE316	DATA BASE MANAGEMENT SYSTEMS LAB	-	-	3	3	50	50	2
CSE317	COMPUTER NETWORKS LAB	-	-	3	3	50	50	2
CSE318	SOFT SKILLS LAB	-	-	3	3	100	0	2
CSE319	QUANTITATIVE& VERBAL APTITUDE I	2	-	1	3	50	0	2
Total		***21***	***5***	***10***	***36***	***450***	***400***	***27***

3rd Year Semester II(Tentative)

CODE NO	SUBJECT NAME	Instruction periods per week				Max Marks		Credits
		Lecture	Tutorial	Lab	Total	Sessional Marks	Semester End Marks	
CSE321	COMPILER DESIGN	4	1	-	5	40	60	4
CSE322	SOFTWARE ENGINEERING	4	1	-	5	40	60	4
CSE323	WEB TECHNOLOGIES	4	1	-	5	40	60	4
CSE324	ADVANCED COMPUTER ARCHITECTURE	4	1	-	5	40	60	4
CSE325	DEPARTMENTAL ELECTIVE I	4	1	-	5	40	60	4
CSE326	OPEN SOURCE TECHNOLOGIES LAB	-	-	3	3	50	50	2
CSE327	SOFTWARE ENGINEERING LAB/MINI PROJECT LAB	-	-	3	3	50	50	2
CSE328	QUANTITATIVE & VERBAL APTITUDE II	2	-	1	3	50	0	2
Total		***22***	***5***	***7***	***34***	***350***	***400***	***26***

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES
(Autonomous)
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
B TECH COURSE PROPOSED SCHEME UNDER AUTONOMOUS SYSTEM 2015-16

4th Year Semester I(Tentative)

CODE NO	SUBJECT NAME	Instruction periods per week				Max Marks		Credits
		Lecture	Tutorial	Lab	Total	Sessional Marks	Semester End Marks	
CSE411	OBJECT ORIENTED ANALYSIS & DESIGN	4	1	-	5	40	60	4
CSE412	CRYPTOGRAPHY & NETWORK SECURITY	3	1	-	4	40	60	3
CSE413	OPEN ELECTIVE II	3	1	-	4	40	60	3
CSE414	DEPARTMENTAL ELECTIVE II	4	1	-	5	40	60	4
CSE415	DEPARTMENTAL ELECTIVE III	4	1	-	5	40	60	4
CSE416	CRYPTOGRAPHY & NETWORK SECURITY LAB	-	-	3	3	50	50	2
CSE417	INDUSTRIAL TRAINING & SEMINAR	-	-	-	0	100	-	2
CSE418	PROJECT PART I			6	6	100	0	4
Total		18	5	9	32	450	350	26

4th Year Semester II(Tentative)

CODE NO	SUBJECT NAME	Instruction periods per week				Max Marks		Credits
		Lecture	Tutorial	Lab	Total	Sessional Marks	Semester End Marks	
CSE421	DEPARTMENTAL ELECTIVE IV	4	1	-	5	40	60	4
CSE422	DEPARTMENTAL ELECTIVE – V	4	1	-	5	40	60	4
CSE423	PROJECT PART II	-	-	12	12	100	100	8
CSE424	MOOC	-	-	-	0	-	-	2
Total		8	2	12	22	180	220	18

Total Credits: **189**

Note:

1. **Academic activities I-** Paper Presentation, Participation in Programming/coding contests.
2. **Academic activities II**- Certificate of participation related to Skill Development Programs/Advanced Topics
3. These courses can be completed from 2nd year to 4th year. It is student choice only.
4. Any one departmental elective, any one Open Elective and MOOC are Optional courses.
5. The total no of credits required to award B Tech Degree :180
6. Students having minimum 8 CGPA with no backlog till 3rd Yr 2nd semester can take advance Departmental elective (IV or/and V) in 4th Yr 1st Semester in place of OPEN ELECTIVE II having registered for OPEN ELECTIVE I in 3rd Yr 1st Semester. Department will conduct extra classes as per the requirement. This is only for students having Internship order. Students should consult the department and decision of Head of the Department is final in this regard.

CSE 211 DATA STRUCTURES AND ALGORITHMS CREDITS: 4
INSTRUCTION: 4Theory & 1Tutorial/ Week SESSIONAL MARKS: 40
FINAL EXAM: 3Hrs FINAL EXAM MARKS: 60

Objectives:

The course should enable the students:
1. To acquire knowledge on several data structures like stacks, queues, linked list, trees and graphs.
2. To have better insight into linear and nonlinear data structures.
3. To learn various sorting and searching techniques.
4. To exercise the applications of data structures.
5. To have a good understanding of problem solving using data structure tools and techniques.

Outcomes:

The student should be able to:
1. Ability to analyze algorithms and determine the time efficiency of each algorithm and their correctness.
2. Master different concepts of abstract data type (ADT) and data structures and their implementations.
3. Ability to determine an appropriate algorithm for a particular problem by analyzing it.
4. Ability to learn different problem solving techniques in addition to the standard ones.

CO-PO Mapping:

S.No	PO a	PO b	PO c	PO d	PO e	PO f	PO g	PO h	PO i	PO j	PO k
CO 1	3	3	2	2	2	0	0	2	0	2	2
CO 2	2	1	2	0	3	0	2	2	2	2	2
CO 3	2	1	3	1	3	1	1	2	3	2	2
CO 4	2	3	2	0	2	0	2	2	3	2	2

COURSE CONTENTS:

UNIT I: 10- 12-Periods

Introduction: Basic Terminology, Elementary Data Organization, Data Structure operations, Fundamentals of algorithmic problem solving – important problem types –Fundamentals of analysis of algorithms and efficiency – Analysis framework – Asymptotic Notations and Basic Efficiency classes – Mathematical Analysis for recursive Algorithms and Non-recursive Algorithms, Algorithm Complexity and Time-Space trade-off.

UNIT II: 10-12 Periods

Arrays: Array Definition, Representation and Analysis, Single and Multidimensional Arrays, address calculation, application of arrays, Character String in C, Character string operation, Array as Parameters, Sparse Matrices.

Stacks: Array Representation and Implementation of stack, Operations on Stacks: Push & Pop, Applications of stack: Conversion of Infix to Prefix and Postfix Expressions, Evaluation of Postfix & Prefix expressions using stack, Recursion, Towers Of Hanoi Problem.

UNIT III 10-12 -Periods

Queues: Array representation and implementation of queues, Operations on Queue: Insert, Delete, Full and Empty. Circular queue, De-queue, and Priority Queue, Applications of Queues.

Linked list: Representation and Implementation of Singly Linked Lists, Traversing and Searching of Linked List, Insertion and deletion to/from Linked Lists, Doubly linked list, Circular Doubly linked list, Implementing priority queue using Linked List, Polynomial Representation using Linked list & addition.

UNIT IV: 10-12-Periods

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

Searching &Sorting: Sequential search, binary search, Interpolation Search, comparison and analysis, Hash Table, Hash Functions. Complexity of Search Algorithm, Insertion Sort, Bubble Sort, Selection sort, Merge Sort.

UNIT V: 10-12 Periods

Graphs: Terminology & Representations- Graphs, Directed Graphs, Adjacency Matrices, Path OR Transitive Closure of a Graph, Warshall's Algorithm, Shortest path Algorithm-Dijkstra's Algorithm, Connected Component and Spanning Trees, Minimum Cost Spanning Trees, Graph Traversals.

TEXT BOOKS

1. Y. Langsam, M. Augenstin and A. Tannenbaum, “Data Structures using C and C++”, Pearson Education, 2nd Edition, 1995.
2. Anany Levitin, “Introduction to the Design and Analysis of Algorithms”, Pearson Education, 3rd Edition, 2012.
3. P. Padmanabham, “C Programming and Data structures”, BS publications, 3rd Edition.

REFERENCE BOOKS

1. E.Horowitz and Sahani, "Fundamentals of Data Structures"
2.Mark Allen Weiss, "Data Structures and Algorithm Analysis in C", Pearson Education, Second Edition.
3. S. Lipschutz, “Data Structures”, McGraw Hill, 1986.
4. P. Dey & M. Ghosh, “Programming in C” , Oxford Univ. Press.
5. ISRD Group, “Data Structures through C++”, McGraw Hill, 2011.

CSE 212 **DIGITAL LOGIC DESIGN** **CREDITS: 3**
(COMMON TO CSE&IT)

INSTRUCTION: 3Theory & 1Tutorial/ Week **SESSIONAL MARKS: 40**
FINAL EXAM: 3Hrs **FINAL EXAM MARKS: 60**

COURSE OBJECTIVES

1. To provide knowledge and understanding of Boolean algebra and digital concepts.
2. To provide the knowledge of analyzing and designing of combinational and sequential logic networks.
3. HDL in this course provides the ability to synthesize the designs in Verilog HDL or VHDL.

COURSE OUTCOMES

By the end of this course, the student should be able to

1. Analyze and synthesize logic circuits by applying the knowledge of number systems, codes and Boolean algebra.
2. Implement new digital designs by synthesizing them using modern engineering tool VHDL.
3. Design and analyze synchronous sequential machines including registers & counters using gates & flip-flops.

CO-PO MAPPING:

	PO-A	**PO-B**	**PO-C**	**PO-D**	**PO-E**	**PO-F**	**PO-G**	**PO-H**	**PO-I**	**PO-J**	**PO-K**
CO-1	3	2	1	1	1	1	2	1	2	2	2
CO-2	1	3	2	1	2	0	3	2	3	2	1
CO-3	2	3	1	1	2	0	1	1	1	3	2

COURSE CONTENTS:

UNIT I

Binary Systems, Boolean Algebra and Logic Gates (10 hrs)

Digital Systems, Binary Numbers, Number Systems, Base Conversion Methods, Complements, Signed Binary Numbers, Binary Codes, Binary Logic.

Basic Definitions, Axiomatic Definition of Boolean Algebra, Basic Theorems and Properties of Boolean Algebra. Boolean Functions, Canonical and Standard Forms, Different Logic Operations, Digital Logic Gates.

UNIT II

Gate-Level Minimization (4 hrs)

The Map Method, Minimal Functions and their properties, Don't-Care Conditions, Tabulation Method, NAND and NOR Implementation, Other Two- Level Implementations, Verilog Hardware Description Language (Verilog HDL).

Combinational Logic Design: **(6 hrs)**

Combinational Circuits, Analysis Procedure, Design Procedure, Design of adders, subtractors, adder-subtractor circuit, BCD adder circuit, applications of adders, Binary Multiplier, Magnitude Comparator, Decoders, Encoders, Multiplexers, Demultiplexers, Verilog HDL for Combinational Circuits.

UNIT III

Sequential Logic Circuits (5 hrs)

Sequential Circuits, Latches, Flip-Flops, Analysis of Clocked Sequential Circuits, Flip-Flop Conversions, Verilog HDL for Sequential Circuits.

Registers and Counters (6 hrs)

Registers, Shift Registers, Ripple Counters, Synchronous Counters, Johnson and Ring counters, Verilog HDL for Registers and Counters.

UNIT IV

Synchronous Sequential Logic (4 hrs)

Basic Design Steps, Serial Adder Example, State Reduction & Assignment Problem.

Fundamentals of Asynchronous Sequential Logic (5 hrs)

Introduction, Analysis Procedure, Design Procedure, circuits with latches, Races and Hazards.

UNIT-V: Programmable Logic Devices (8 hrs)

Programmable Logic Devices : PROM, PLA, PAL, realization of switching functions using PROM, PLA and PAL; comparison of PROM, PLA and PAL, Programming tables of PROM, PLA and PAL, Sequential Programmable Devices.

TEXT BOOKS

1. M. Morris Mano, “Digital Design”, Pearson Education, Inc., 2008 , 4th Edition.

REFERENCE BOOKS

1**.** Zvi Kohavi, “Switching and Finite Automata Theory”, Tata McGraw-Hill,1978,2nd Edition.

2. Frederick, “Introduction to Switching Theory and Logical Design”, 2011 & J. Hill and Gerald R. Peterson, John Wiley and Sons, 2011,3rd Edition.

3. William I. Fletcher, “An Engineering Approach to Digital Design”, PHI, 2008.

CSE 213 DISCRETE MATHEMATICAL STRUCTURES CREDITS: 4
(COMMON TO CSE& IT)

INSTRUCTION: 4 Theory & 1Tutorial/ Week SESSIONAL MARKS: 40
FINAL EXAM: 3Hrs FINAL EXAM MARKS: 60

Course Objective :

The knowledge of Mathematics is necessary for a better understanding of almost all the Engineering and Science subjects. Here our intention is to make the students acquainted with the concept of basic topics from Mathematics, which they need to pursue their Engineering degree in different disciplines.

Course Outcomes:

At the end of the course student should be able to:

CO - 1	Understand set theory, relations, mathematical logic, mathematical reasoning and to study about the validity of the arguments.
CO - 2	Apply basic counting techniques to solve combinatorial problems.
CO - 3	Understand Recurrence Relation, Generating functions and solving problems involving recurrence equations.
CO - 4	Familiarize the different types of binary relations and related algorithms on transitive closure.
CO - 5	Familiarize with the applications of graphs, trees and algorithms on minimal spanning tress.

Mapping of course outcomes with program outcomes:

Course Outcomes	PO-a	PO-b	PO-c	PO-d	PO-e	PO-f	PO-g	PO-h	PO-i	PO-j	PO-k
CO - 1	**3**								**1**		**3**
CO - 2	**3**								**1**		**3**
CO - 3	**3**								**1**		**3**
CO - 4	**3**								**1**		**3**
CO - 5	**3**								**1**		**3**

COURSE CONTENTS:

UNIT-I

Mathematical Logic (15 Periods)

Sets-Operations on sets-relations-functions-Fundamentals of Logic- Logical inferences-Methods of proof of an implication-First Order logic and Other methods Proof -Rules of inference for quantified Propositions –Mathematical Induction.

UNIT II

Elementary Combinatorics (08 Periods)

Basics of Counting- Combinations and Permutations-Their Enumeration with and without repetition-Binomial coefficients-Binomial and Multinomial Theorems-The Principle of Inclusion-Exclusion.

UNIT III

Recurrence Relations (08 Periods)

Generating Functions of Sequences-Calculating their Coefficients-Recurrence relations-Solving recurrence relations-Method of characteristic Roots- Non-homogeneous Recurrence relations and their solutions.

.

UNIT IV

Relations and Digraphs (09 Periods)

Relations and Directed Graphs - Special Properties of Binary relations- Equivalence Relations-Ordering Relations-Lattices and Enumeration- Operations on relations-Paths and Closures-Directed Graphs and Adjacency matrices .

UNIT V

Graphs (20 Periods)

Introduction to Graphs – types of Graphs – Graphs basic terminology and special types of simple graphs – representation of graphs and graph isomorphism – Euler paths and circuits- Hamilton paths and circuits – Planar graphs – Euler's formula.

Introduction to Trees and their properties – Spanning Trees – Depth First Search , Breadth First Search – Minimum Spanning Trees – Kruskal's Algorithm and Prim's Algorithm.

TEXT BOOKS:

1). Joe L. Mott, Abraham Kandel & T. P. Baker, "Discrete Mathematics for computer scientists & Mathematicians" Prentice Hall of India Ltd, New Delhi.

REFERENCE BOOKS:

1) Keneth. H. Rosen, "Discrete mathematics and its applications", Tata McGraw- Hill Publishing Company, New Delhi
2) Richard Johnsonbaug, "Discrete mathematics" by Pearson Education, New Delhi.

CSE 214 **Object oriented Programming with JAVA** **CREDITS: 3**
INSTRUCTION: 3Theory & 1Tutorial/ Week **SESSIONAL MARKS: 40**
FINAL EXAM: 3Hrs **FINAL EXAM MARKS: 60**

Prerequisites:

Basic knowledge of computer fundamentals
Student must have knowledge of some programming languages (such as C, C++)

Course Objectives:

- To understand object oriented programming concepts, and apply them in problem solving.
- To learn the basics of java Console and GUI based programming.

COURSE OUTCOMES:

CO-1: Understanding of OOP concepts and basics of Java programming (Console and GUI based).

CO-2: The skills to apply OOP and Java programming in problem solving.

CO-3: Should have the ability to extend his/her knowledge of Java programming further on his/her own.

CO-PO MAPPING:

	PO-A	PO-B	PO-C	PO-D	PO-E	PO-F	PO-G	PO-H	PO-I	PO-J	PO-K
CO-1	3	3	3	2	2	0	2	2	2	2	2
CO-2	3	3	3	2	2	0	2	2	2	2	2
CO-3	2	2	1	3	2	0	2	3	2	2	2

COURSE CONTENTS:

UNIT-I **10-12hours**

OOP concepts - Data abstraction, encapsulation, inheritance, benefits of inheritance, polymorphism, classes and objects, Procedural and object oriented programming paradigms.

Java programming - History of Java, data types, variables, operators. Control structures, arrays, console input and output, formatting output. Simple programs on java.

Introduction to Classes, objects, constructors, methods, parameter passing, static fields and methods, access control, this reference, overloading constructors and methods, recursion, final keyword, garbage collection, finalize method, inner class and uses of inner classes, String handling.

UNIT-II **10-12 hours**

Inheritance – Basics, using super keyword, multilevel hierarchy, Member access rules, preventing inheritance- using final, the Object class and its methods.

Polymorphism - dynamic binding, method overriding, abstract class and methods.

Interfaces - Interfaces vs. Abstract class, defining an interface, implementing interfaces, accessing implementations through interface references, extending interfaces.

Packages - Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages.

UNIT –III **10-12 hours**

I/O: I/O basics, byte and character streams , read/ write console input/output, reading and writing files.

Exception handling – Fundamentals, Exception types, use of try and catch, throw, throws, finally, multiple catches, built-in exceptions, user defined exceptions.

Multithreading – Thread Priorities, synchronization, messaging, reading a thread, creating multiple threads, use of alive and join, inter-thread communication- suspending resuming and stopping threads, producer-consumer problem with multithreading.

UNIT-IV **10-12 hours**

Applets: Basics, Applet class, Applet Architecture, Applet skeleton. The HTML Applet tag, A simple banner applet. Difference between Application program and applet program.

AWT-Working with Windows, Graphics and Text: AWT Classes, Window fundamentals, working with Frame windows, creating a frame window in an applet. Working with graphics, working with colors, working with fonts.

UNIT-V **10-12 hours**

Event Handling: The Delegation event model, Event classes, Event Listener interfaces, handling mouse and keyboard events.

Using AWT Controls, Layout Managers and Menus: Control fundamentals, Labels, Buttons, Check Boxes, Choice Controls, Lists, Scroll bars, Text field, Text Area, Layout Managers.

TEXT BOOKS

1. Herbert Schildt, "JAVA The Complete Reference", TataMcGraw Hill,seventh edition.
2. Y.Daniel Liang (PHI),"Introduction to JAVA PROGRAMMING"

REFERENCES BOOKS

1. P.J. Deitel and H.M. Deitel, "Java for Programmers", Pearson education (OR) P.J. Deitel and H.M. Deitel, "Java: How to Program", PHI.
2. P. Radha Krishna, "Object Orientd Programming through Java", Universities Press.
3. Bruce Eckel, "Thinking in Java", Pearson Education
4. Bruce Eckel, "Programming in Java", Pearson Education
5. S. Malhotra and S. Choudhary, "Programming in Java", Oxford Univ. Press.

CSE 215 PROBABILITY, STATISTICS AND QUEUING THEORY CREDITS: 4

INSTRUCTION: 4Theory & 1Tutorial/ Week **SESSIONAL MARKS: 40**
FINAL EXAM: 3Hrs **FINAL EXAM MARKS: 60**

Course Objective :

The knowledge of Mathematics is necessary for a better understanding of almost all the Engineering and Science subjects. Here our intention is to make the students acquainted with the concept of basic topics from Mathematics, which they need to pursue their Engineering degree in different disciplines.

Course Outcomes:

At the end of the course student should be able to:

CO - 1	Understand the concepts of various statistical measures like mean, variance and standard deviation of a random variable.
CO - 2	Familiarize the different types of probability distributions and their properties.
CO - 3	Compute simple correlation between the variables and fit straight line, parabola by the principle of least squares.
CO - 4	Analyze the statistical data and apply various small or large sample test for testing the hypothesis.
CO - 5	Learn about different Queuing models and its applications.

Mapping of course outcomes with program outcomes:

Course Outcomes	PO-a	PO-b	PO-c	PO-d	PO-e	PO-f	PO-g	PO-h	PO-i	PO-j	PO-k
CO - 1	**3**								**1**		**3**
CO - 2	**3**								**1**		**3**
CO - 3	**3**								**1**		**3**
CO - 4	**3**								**1**		**3**
CO - 5	**3**								**1**		**3**

COURSE CONTENTS:

UNIT I

Probability & Mathematical Expectations (12 Periods)

Introduction to probability: Definition of Random Experiment, Events and Sample space, Definition of probability, Addition and Multiplication theorems, Conditional probability, Baye's Theorem, Simple Problems on Baye's theorem. Random Variables: Discrete and Continuous random variables, Distribution function of random variable, Properties, Probability mass function, Probability density function, Mathematical expectation, Properties of Mathematical expectations, Mean and Variance.

UNIT II

Probability Distribution (14 Periods)

Discrete Distributions: Binomial Distribution, Mean and Standard Deviations of Binomial Distribution, Poisson distribution, Mean and Standard Deviations of Poisson Distribution, Applications. Continuous Probability Distributions: Uniform Distribution, Exponential Distribution, Normal Distribution, Properties of Normal Distribution, Importance of Normal Distribution, Area properties of Normal curve.

UNIT III

Curve Fitting , Correlation and Regression (10 Periods)

Curve Fitting : Principle of Least Squares , Method of Least Squares (Straight Line and Parabola).

Correlation : Definition, Measures of correlation, Correlation for Bivariate Distribution, Rank correlation coefficients.

Regression : Simple linear regression, regression lines and properties.

UNIT IV

Testing of Hypothesis (14 Periods)

Formulation of Null Hypothesis, Critical Region, Level of Significance.

Small Samples : Students t - distribution (Significance test of a sample mean, Significance test of difference between sample means), F- distribution, χ^2 - test, Goodness of fit.

Large samples : Test of Significance of Large Samples – Single Proportion, Difference between two Proportions , Single mean and Difference of means.

UNIT V

Queuing Theory (10 Periods)

Queue description, characteristics of a queuing model, study state solutions of M/M/1: α Model, M/M/1 ; N Model.

TEXT BOOKS

1. T.Veerarajan, "Probability, Statistics and Random Processes" Tata McGraw Hill Publications.

REFERENCE BOOKS

1. Kishor S. Trivedi , "Probability & Statistics with Reliability, Queuing and Computer Applications" Prentice Hall of India ,1999 .

CSE 216 **Data Structures Lab** **Credits: 2**

Instruction: 3 Periods/week Sessional Marks: 50

End. Exam: 3 Hours End-Exam-Marks: 50

Outcomes of the Lab:

1. Be able to design and analyze the time and space efficiency of the data structure.
2. Be capable to identify the appropriate data structure for given problem.
3. Have practical knowledge on the application of data structures.
4. Able to apply different problem solving techniques on real world problems world problem.

CO-PO Mapping:

S.No	PO a	PO b	PO c	PO d	PO e	PO f	PO g	PO h	PO i	PO j	PO k
CO 1	3	3	2	2	2	0	0	2	0	2	2
CO 2	2	1	2	0	3	0	2	2	2	2	2
CO 3	2	1	3	1	3	1	1	2	3	2	2
CO 4	2	3	3	0	2	0	2	3	3	2	2

List of Programs:

1) Develop C programs to implement the following using an array.
   a) Stack b) Queue

2) Develop C programs to implement the following using a singly linked list.
a) Stack b) Queue

3) Develop a C program to do the following
 a) Infix to Postfix conversion. b) Evaluation of postfix expression.

4) Develop C programs to implement the following
   a) Circular Queue b) Priority Queue

5) Implement the dequeue (double ended queue) using a doubly linked list and an array.

6) Write a C program to perform the following operations:
a) Insert an element into a binary search tree.
b) Delete an element from a binary search tree.
c) Search for a key element in a binary search tree.

7) Write C programs that use non-recursive functions to traverse the given binary tree in
a) Pre-order b) In-order c) Post-order.

8) Write C programs for the implementation of BFS and DFS for a given graph.

9) Write C programs for implementing the following sorting methods:
a) Merge sort b) Quick sort

10)Write C programs for implementing the following searching methods:
a) Linear Searchb) Binary search

11)Write a C program to store k keys into an array of size n at the location computed using a hash function, loc = key % n, where k<=n and k takes values from [1 to m], m>n.

12) Write a C program to handle the collisions using the following collision resolution Technique
a) Linear probing b) Quadratic probing c)Separate Chaining

Note: All programs are to be implemented in C only

TEXT BOOKS

1. Y. Langsam, M. Augenstin and A. Tannenbaum, "Data Structures using C" Pearson Education, 2nd Edition, 1995.
2. Richard F, Gilberg ,Forouzan, "Data Structures", Cengage, 2005,2/e.

CSE 217 **Digital Electronics Lab** **Credits: 2**
(Common to CSE and IT)

Instruction: 3 Periods/week Sessional Marks: 50
End. Exam: 3 Hours End-Exam-Marks: 50

The following are the list of laboratory experiments for DIGITAL ELECTRONICS Laboratory in 2-1 (CSE & I.T Dept Autonomous) for the academic year (2016-17).

***NOTE:** FOUR Experiments from each cycle should be done compulsorily.

CYCLE-I:

1. Study of passive, active components & Integrated Circuits.
2. To study the regulation characteristics of given Integrated Circuits.
3. To verify the adder operation & subtractor operation using Operational amplifiers.
4. To verify the truth tables of given Logic Gates.

CYCLE-II:

1. Verification of truth tables of Logic gates using IC's.
2. Design a combinational circuit for Code Converters using IC's.
3. Design a combinational circuit for Adders & Subtractors (HA & FA) using IC's.
4. Design a sequential circuit for Flip-Flop and verify its characteristics using IC's..
5. Design a bidirectional Universal Shift Register Using IC74LS194.
6. Design of Counters using IC74LS73.

CYCLE-III: (Simulation using VHDL)

1. Write a program for verification of Basic Gates.
2. Write a program for Adder & Subtractor.
3. Write a program for Flip Flops.
4. Write a program for MUX & DEMUX.
5. Write a Program for Shift Registers.

CSE 218 **JAVA Lab** **CREDITS: 2**
INSTRUCTION: 3 Hrs/ Week **SESSIONAL MARKS: 50**
FINAL EXAM: 3Hrs **FINAL EXAM MARKS: 50**

Prerequisites:

Basic knowledge of computer fundamentals
Student must have knowledge of some programming languages (such as C ,C++)

Course Objectives:

- To understand object oriented programming concepts, and apply them in problem solving.
- To learn the basics of java Console, GUI based programming and networking programming.

COURSE OUTCOMES:

CO-1: Understanding of OOP concepts and basics of Java programming (Console and GUI based).

CO-2: The skills to apply OOP and Java programming in problem solving.

CO-3: Should have the ability to extend his/her knowledge of Java programming further on his/her own.

CO-PO MAPPING:

	PO-A	PO-B	PO-C	PO-D	PO-E	PO-F	PO-G	PO-H	PO-I	PO-J	PO-K
CO-1	3	3	3	2	2	0	2	2	2	2	2
CO-2	3	3	3	2	2	0	2	2	2	2	2
CO-3	2	2	1	3	2	0	2	3	2	2	2

.

List of Programs:

1. Write a program to find the factorial of a given number.
2. Write a program to print numbers in sorting order.
3. Create a class Odometer that displays the number of kilometers a vehicle run. Give samples as trip information like number of kilometers travelled, fuel consumption per litre. The task is to find the mileage of the vehicle running at different samples of trip information.
4. Create a class Day that represents day, month and year of the calendar day. The class Day should be able to accept the date, update the date, delete the date from a calendar list of activities. Create a class Time that represents hours, minutes, seconds of a clock. The class Time should accept the time, update the time, delete the time from a list of events created for a day using the Day Class.
5. Write a program on illustration of use of packages.
6. Write a program to implement interfaces.
7. Write a program that implements a stack ADT that converts infix expression into postfix expression
8. Write a program to read a file and displays the file on the screen within line number before each line
9. Write a program to copy contents of a file into another file using File streams.
10. Write a program for handling ArrayIndexOutofBoundsException and Divide-by-zero Exception.
11. Write a program for custom exception creation.
12. Write a program on multi-threading showing how CPU time is shared among all the threads.
13. Write a program for Producer-Consumer problem using threads.
14. Write an applet that displays a simple message.
15. Write an applet to handle the mouse events and keyboard events.
16. Write a program to develop a simple calculator. Using Grid layout arrange buttons for the digits and +,-,* % operations. The computation should be performed with a button click "Compute". Display the result on a text field.

CSE 221 DATA COMMUNICATIONS CREDITS: 4

INSTRUCTION: 4 Theory & 1Tutorial/ Week SESSIONAL MARKS: 40
FINAL EXAM: 3Hrs FINAL EXAM MARKS: 60

Prerequisite:

Basic knowledge of Computer Hardware, Network basics.

COURSE OBJECTIVES:

- To educate concepts, vocabulary and techniques currently used in the area of Data Communication, Networking and Internet.
- To interpret the Digital encoding Techniques in Data Communication.
- Familiarize the student with the basic taxonomy and terminology of the Data and signals, Signal Transmission, and Transmission Impairments.
- To accumulate existing state-of-the-art in Data Link Layer concepts and sliding window protocols and its applications.
- To analyze the functions of physical layer and gain knowledge in different mediums used for data transfer.
- Introduce the student to illustrate the point in Data Communication & networking concepts, preparing the student for that entry level courses.

COURSE OUTCOMES:

1	Fundamental understanding of data and signals, and formulate the data communication models using OSI and TCP/IP protocol architectures in data communications
2	Able to explain the Characteristics of conversion of analog/digital data into digital/analog signals, with respect to their time domain and frequency domain and bandwidth requirements
3	Able to understand the difference among transmission impairments, including attenuation, distortion and noise.
4	Able to understand and explain the flow control methods and error control methods for data link layers and also explain how does HDLC protocol works.
5	Able to Understand Data communication Hardware and communication Software in such a way that how these Multiplexer, Concentrator, Front-End Processor, Cluster Control Units operates.

CO-PO mapping

POA	POB	POC	POD	POE	POF	POG	POH	POI	POJ	POK
2	1	2	1	1	3	1	2	2	1	1
1	2	2	1	3	1	2	1	0	1	0
2	1	2	0	1	2	1	1	2	2	0
2	1	3	2	3	2	2	1	2	2	3
1	2	2	3	3	2	2	3	2	1	4

UNIT 1:

Data Communications, Data Networking, Internet: A Communications Model, Data Communications, Networks, The Internet, An Example Configuration, Protocol Architecture. The Need for a Protocol Architecture: The TCP/IP Protocol Architecture, The OSI Model, Traditional Internet-Based Applications,Characteristics of Data,Transmission: Concepts and Terminology, Analog and Digital Data Transmission, Transmission Impairments.

UNIT 2:

Transmission Media:

Guided Transmission Media, Wireless Transmission Data Encoding, Digital Data, Digital Signals, Analog Signals, Analog.

UNIT 3:

The Digital Data Communication Techniques:

Asynchronous and Synchronous Transmission, Line Configurations, Interfacing.Data Link Control Flow Control, Types of Errors, Error Detection, Error Control, High-Level Data Link Control (HDLC).

UNIT 4:

Local Area Network:

Overview, LAN Protocol Architecture, Bridges, Layer 2 and Layer 3 Switches.
High-Speed LANs: The Emergence of High-Speed LANs.Wireless LANs:Overview,Wireless LAN Technology,IEEE 802.11 Architecture and Services.

UNIT 5:

Modems and Modem Circuits. Multiplexing: Frequency-Division Multiplexing, Synchronous Time-Division Multiplexing: Characteristics, TDM Link Control, Digital Carrier Systems Statistical Time-Division Multiplexing: Characteristics, The Concept of Spread Spectrum,Frequency Hopping Spread Spectrum,Direct Sequence Spread Spectrum,Code-Division Multiple Access.

TEXT BOOKS

William Stallings, "Data and Computer Communications" , Pearson Education Inc., 2010 8th Edition.

REFERENCE BOOKS

Behrouz A. Forouzan, "Data Communications and Networking", TMH, 2004, 3rd Edition.

CSE 222 MICROPROCESSOR AND INTERFACING CREDITS: 3
INSTRUCTION: 3Theory & 1Tutorial/ Week SESSIONAL MARKS: 40
FINAL EXAM: 3Hrs FINAL EXAM MARKS:60

PREREQUISITE: Digital Logic

Course Objectives:

1. The objective of this course is to become familiar with the architecture and the instruction set of an Intel microprocessor.
2. Assembly language programming will be studied as well as the design of various types of digital and analog interfaces
3. To provide solid foundation on interfacing the external devices to the processor according to the user requirements to create novel products and solutions for the real time problems.
4. To assist the students with an academic environment aware of excellence guidelines and lifelong learning needed for a successful professional career.
5. The accompanying lab is designed to provide practical hands-on experience with microprocessor software applications and interfacing techniques

Course Outcomes:

co-1	To develop an in-depth understanding of the operation of microprocessors .
co-2	Understanding various pins and its functions , the instruction sets and timing sequences of 8085 and 8086 processor
co-3	To illustrate the Interfacing with Different I/O Devices.
co-4	To implement interfacing from a microprocessors based system to peripheral devices. To understand software/ hardware interrupts and further write programs to perform I/O using handshaking and interrupts

CO-PO Mapping

	PO-a	PO-b	PO-c	PO-d	PO-e	PO-f	PO-g	PO-h	PO-i	PO-j	PO-k
CO-1	3	1	0	0	0	0	0	0	0	1	0
CO-2	3	3	2	2	0	0	1	1	1	2	0
CO-3	3	3	2	0	3	0	2	0	2	2	0
CO-4	2	1	1	0	2	0	1	0	2	2	0

UNIT I

The 8085A µP. Architecture and Instruction Set:

Introduction to Microprocessors and Microcomputers, Internal Architecture and Functional/Signal Description of typical 8-bit µP.- 8085, Instruction Set and Timing Diagrams of 8085 µP. Interfacing SRAMs, and EPROMs to 8085.

15 h

UNIT II

Programming the 8085 µP.:

Assembly Language Programming Requirements, Programming Techniques: Looping, Counting, and Indexing, Counter and timing Delays, Stack and Subroutines, Code Conversion, BCD Arithmetic, 16-bit data Operations, Interrupts and Interrupt Service Routines

10h

UNIT III

Interfacing Peripheral ICs to Intel 8085

Parallel I/O Interface - 8255, Serial I/O Interface – 8251, Timer Interface - 8253, Keyboard/Display Interface - 8279, Interrupt Controller Interface - 8259,D/A Conversion Methods, A/D Conversion methods, Interfacing DAC, Interfacing ADC.

20h

UNIT IV

The 8086 µP. Architecture and.:

Internal Architecture and Functional/Signal Description of 8086/8088
Segmented Memory, Maximum-Mode and Minimum-Mode Operation, Addressing Modes.

10h

UNIT V

Programming the 8086 µP

Instruction Set and Timing Diagrams Assembly Language Requirements, Data Definition, Loops Procedures, Modular programming, and Macros

5h

TEXT BOOKS:

1. Ramesh S. Gaonkar, "Microprocessor Architecture, Programming, and Applications with the 8085" Penram International ,6th Edition,
2. John E.Uffenbeck, "The 80x86 Family, Design, Programming and Interfacing3rdEdition, Pearson Education Inc.", 2002

REFERENCE BOOKS:

1. BARRY B. BREY, "The Intel Microprocessors 8086/8088, 80186/80188,80286,80386 and 80486, Pentium, Pentium Pro Processor, Pentium II, Pentium III, Pentium 4, Architecture, Programming and Interfacing", Pearson Education Inc., 2003,6thEdition.
2 Walter A. Tribel and Avtar Singh," The 8088 and 8086 Microprocessors, Programming, interfacing, Software, Hardware, and Applications", Pearson Education Inc., 2003,4thEdition.
3. Douglass V. Hall,"Microprocessors and Interfacing, Programming and Hardware" , TMH Edition, 1999, 2ndEdition
4. Sanjay K Bose, "Hardware and Software of Personal Computers", New Age International (P) Ltd., 1991

CSE 223 OPERATING SYSTEMS CREDITS: 4

INSTRUCTION: 4Theory & 1Tutorial/ Week SESSIONAL MARKS: 40
FINAL EXAM: 3Hrs FINAL EXAM MARKS: 60

Course Objectives:

1. Introduce various fundamental concepts and principles of operating systems .
2. This course provides a comprehensive introduction to understand the underlying techniques and approaches which constitute a coherent body of knowledge in operating systems. In particular, the course will consider inherent functionality and processing of program execution .
3. The emphasis of the course will be placed on understanding how the various elements that underlie operating system interact and provides services for execution of application software

Course Outcomes:

1. An appreciation of the role of an operating system.
2. Understand the theory and logic behind the design and construction of operating systems.
3. Examine the algorithms used for various operations on operating systems.
4. Differentiate between various operating systems functionalities in terms of performance.
5. Know the problems in the design of operating system and study the probable solutions.
6. Become aware of the issues in the management of resources like processor, memory and Input-output.

Mapping of COs and POs

	POA	POB	POC	POD	POE	POF	POG	POH	POI	POJ	POK
CO1	2	-	3	-	3	-	-	3	2	3	2
CO2	3	-	3	-	2	-	-	3	2	3	-
CO3	2	2	-	-	3	-	-	-	3	-	-
CO4	3	1	-	-	-	-	2	-	2	-	-
CO5	2	2	-	2	3	-	-	3	3	-	-
CO6	2	-	-	-	3	-	2	-	-	1	1

COURSE CONTENTS:

UNIT I

Introduction to OS

Introduction to operating systems – review of computer organization – operating system structures – system calls – system programs – system structure – virtual machines.

Process Management

Processes: Process concept – Process scheduling – Operations on processes –Cooperating processes – Interprocess communication. Multi threaded programming. Communication in client-serversystems. Multi-Threaded Programming:Overview; Multithreading models; Thread Libraries; Threading issues.

UNIT II

Process Scheduling and Synchronization

CPU Scheduling: Scheduling criteria – Scheduling algorithms – Multiple-processorscheduling – Real time scheduling – Algorithm Evaluation. Process Synchronization: The critical-section problem –Synchronization hardware – Semaphores – Classic problems of synchronization – critical regions – Monitors. Deadlock: System model – Deadlock characterization –Methods for handling deadlocks – Deadlock prevention – Deadlock avoidance, Deadlock detection – Recovery from deadlock.

UNIT III

Memory Management

Memory Management: Background – Swapping – Contiguous memory allocation – Paging – Segmentation – Segmentation with paging. Virtual Memory: Background – Demand paging – Process creation – Page replacement – Allocation of frames – Thrashing.

UNIT IV

File Systems and its Implementation

File-System Interface: File concept – Access methods – Directory structure – Filesystem mounting – Protection. File-System Implementation : Directory implementation – Allocation methods – Free-space management – efficiency and performance – recovery – log-structured file systems.

UNIT V

Secondary Storage Structures and Protection

Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management; Swap space management. Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights, Capability-Based systems.

Case Study: The Linux Operating System: Linux history; Design principles; Kernel modules; Process management; Scheduling; Memory management; File systems, Input and output; Inter-process communication

TEXT BOOKS

1. Silberschatz, Galvin, and Gagne, “Operating System Concepts”, Wiley India Pvt Ltd, 2003, Sixth Edition.

REFERENCES

1. Andrew S. Tanenbaum, “Modern Operating Systems”, Pearson Education, 2004, Second Edition.
2. Gary Nutt, “Operating Systems”, Pearson Education, 2004 ,Third Edition.
3. Harvey M. Deitel, “Operating Systems”, Pearson Education, 2004, Third Edition.

CSE 224 COMPUTER ORGANIZATION CREDITS: 4

INSTRUCTION: 4Theory & 1Tutorial/ Week SESSIONAL MARKS: 40
FINAL EXAM: 3Hrs FINAL EXAM MARKS: 60

PREREQUISITE: Digital Logic

COURSE OBJECTIVE:

- To understand the basics of computer hardware and how software interacts with computer hardware.
- To understand the structure, function and characteristics of computer systems.
- To understand the basic structure and operation of digital computer.
- To study the design of arithmetic and logic unit.
- To study the two types of control unit techniques and the concept of pipelining.
- To understand the hierarchical memory system including cache memories and virtual memory.
- To understand the different ways of communicating with I/O devices and standard I/O interfaces.

COURSE OUTCOMES:

CO-1: The student will be familiar with the basic knowledge of the design of digital logic circuits and apply to computer organization.
CO-2: The student will be familiar with the functional units of the processor such as the register file and arithmetic-logical unit.
CO-3: The student will learn with the representation of data, addressing modes, instructions sets.
CO-4: The student will be able to describe the organization of digital computers and explain the basic principles and operations of different components.
CO-5: The student will be able to evaluate the performance of CPU, memory and I/O operations.

CO-PO MAPPING:

	PO-A	PO-B	PO-C	PO-D	PO-E	PO-F	PO-G	PO-H	PO-I	PO-J	PO-K
CO-1	3	2	1	1	1	1	2	1	2	2	3
CO-2	1	3	2	1	2	2	3	2	3	2	3
CO-3	2	3	1	1	2	2	1	1	1	3	3
CO-4	2	2	1	2	2	3	3	2	2	3	2
CO-5	2	3	3	2	1	2	3	3	2	3	3

COURSE CONTENTS:

UNIT-1

Register Transfer and Micro operations :

Register Transfer Language, Bus and Memory Transfers, Arithmetic, Logic and Shift Micro operations, Arithmetic Logic Shift Unit,

Computer Arithmetic:

Introduction, Addition and Subtraction, Booth Multiplication Algorithm, Decimal Arithmetic Unit.

UNIT-2

Basic Computer Organization:

Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory-Reference Instructions, Input-Output and Interrupt, Complete Computer Description.

UNIT-3

Control Design:

Hardwired & Micro Programmed (Control Unit), Control Memory, Address Sequencing, Conditional and Unconditional Branching, Micro program Example.

UNIT-4

Central Processing Unit:

Introduction, General Register Organization, Stack Organization, Instruction Formats, Addressing Modes with numerical examples, Data Transfer and Manipulation, Program Control, Program Interrupt, Types of interrupts, CISC Characteristics, RISC Characteristics.
Introduction to Parallel Processing, Pipelining – General Considerations.

UNIT-5

Input-Output Organization:

Peripheral Devices, Input-Output Interface, Asynchronous Data Transfer, Modes of Transfer, Priority Interrupt, Direct Memory Access.

Memory Organization:

Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory.

TEXT BOOKS

1. M.Morris Mano, "Computer System Architecture", Pearson Education Inc., 2003, Third Edition,.

REFERENCE BOOKS

1. John D. "Carpinelli ,Computer Systems Organization and Architecture", Pearson Education Inc., 2003.

CSE 225 FORMAL LANGUAGES AND AUTOMATA THEORY CREDITS: 4

INSTRUCTION: 4Theory & 1Tutorial/ Week SESSIONAL MARKS: 40
FINAL EXAM: 3Hrs FINAL EXAM MARKS: 60

Course Objectives:

- Introduce concepts in automata theory and theory of computation
- Identify different formal language classes and their relationships
- Design grammars and recognizers for different formal languages
- Prove or disprove theorems in automata theory using its properties
- Determine the decidability of computational problems.

Course Outcomes:

1. Understand basic properties of formal languages and formal grammars
2. Understand basic properties of deterministic and nondeterministic finite automata
3. Understand the relation between types of languages and types of finite automata
4. Understand basic properties of Turing machines and computing with Turing machines
5. Understand the concepts of decidability, the concepts of NP-completeness and NP-hard problems
6. Understand the challenges for Theoretical Computer Science and its contribution to other sciences

Mapping of COs and Pos

	POA	POB	POC	POD	POE	POF	POG	POH	POI	POJ	POK
CO1	2	--	-	-	2	-	1	-	1	2	2
CO2	2	-	-	-	2	-	1	-	1	2	2
CO3	2	-	-	-	2	-	1	-	1	2	2
CO4	2	-	-	-	2	-	1	-	1	2	2
CO5	2	-	-	-	2	-	1	-	1	2	2
CO6	3	-	-	-	2	-	1	-	1	2	2

COURSE CONTENTS:

UNIT -1

Introduction to Finite Automata: Introduction to Finite Automata; The Central concepts of Automata theory; Deterministic finite automata;Nondeterministic finite automata.

Finite Automata, Regular Expressions: An application of finite automata ;Finite automata with Epsilon-transitions; Regular expressions; FiniteAutomata and Regular Expressions; Applications of Regular Expressions. Two way finite automata, finite automata with output**:** Mealy and Moore machines.

UNIT -2

Regular Languages, Properties of Regular Languages: Regular languages; Proving languages not to be regular languages; Closure propertiesof regular languages; Decision properties of regular languages; Equivalence and minimization of automata. Pumping lemma, closure properties, decision algorithm, MyHill- Nerode theorem and minimization of finite automata.

UNIT -3

Context-Free Grammars And Languages : Context –free grammars; Parse trees; Applications; Ambiguity in grammars and Languages

UNIT -4

Pushdown Automata: Definition of the Pushdown automata; the languages of a PDA; Equivalence of PDA's and CFG's; Deterministic PushdownAutomata.

Properties of Context-Free Languages: Normal forms for CFGs; The pumping lemma for CFGs; **Closure properties of CFLs**

UNIT -5

Introduction To Turing Machine: Problems that Computers cannot solve;The turing machine; Programming techniques for Turning Machines;Extensions to the basic Turing Machines; Turing Machine and Computers.church's hypothesis. The classes P and NP; NP-Completeness; Satisfiability and Cook's theorem; Polynomial reduction and some NP-complete problems.

Undecidability: properties of recursive and recursively enumerable languages, universal Turing machines, Rice's theorem, Post Correspondence Problem, Greibach's theorem, introduction to recursive function theory, Oracle computation; Chomsky Hierarchy**:** regular grammars, unrestricted grammars, context sensitive languages, relations between classes of languages.

TEXT BOOKS

1.John E. Hopcroft, Rajeev Motwani, Jeffrey D.Ullman: “Introduction to Automata Theory, Languages and Computation”, Pearson Education, 2007,3rd Edition.

REFERENCE BOOKS

1. Mishra &Chandrasekharan, “Theory of computer science: Automata language and computation”, Prentice Hall of India, 3rd Ed,2007.

2. K.L.P. Mishra: “Theory of Computer Science, Automata, Languages,
and Computation”, PHI Learning, 2009,3rd Edition.

3. John C Martin: “Introduction to Languages and Automata Theory”, Tata McGraw-Hill, 2007 3rd Edition.

4. P. Linz, “Introduction to Formal Language and Computation”, Narosa, 2nd Ed, 2006.

CSE 226 MICROPROCESSOR AND INTERFACING LAB CREDITS: 2
INSTRUCTION: 3Periods/ Week SESSIONAL MARKS: 50
FINAL EXAM: 3Hrs FINAL EXAM MARKS: 50

Course Objectives:

1.Developing of assembly level programs and providing the basics of the processors
2. To provide solid foundation on interfacing the external devices to the processor according to the user requirements to create novel products and solutions for the real time problems
3. To assist the students with an academic environment aware of excellence guidelines and lifelong learning needed for a successful professional career

Course Outcomes:

CO-1:Able to understand the problem and interfacing of peripheral devices through ALP programming .
CO-2: The students will learn how to design, build, and debug simple microcontroller based systems.
CO-3:To be able to test a solution for different parameters and cases and analyze the solution
CO-4:The students will work in groups of 2 to 4 and thereby learn how to cooperate in teams.

CO-PO Mapping

	PO-a	PO-b	PO-c	PO-d	PO-e	PO-f	PO-g	PO-h	PO-i	PO-j	PO-k
CO-1	0	2	3	0	1	0	0	2	1	3	2
CO-2	0	3	3	0	1	0	0	2	1	3	2
CO-3	1	3	0	0	1	0	0	2	1	3	2
CO-4	0	0	0	3	1	0	0	2	0	0	0

Assembly Language Programming :

1. 8085 Assembly Language Programming according to theory course using the following trainers : Keyboard Monitor of 8085µP Trainer.

3 Weeks

2.INTERFACING WITH 8085 TRAINER

2.1.8255 study card for mode 0,1 practice.
2.2 HEX KEYBOARD AND DOT MATRIX HEX LED DISPLAY INTERFACE
2.3 8279-PROGRAMMABLE KEYBOARD/DISPLAY INTERFACE

3 weeks

3.INTERFACING WITH PC

3.1 STEPPER MOTOR CONTROLLER
3.2 DAC/ADC INTERFACE
3.3 8253 TIMER INTERFACE
3.4 TRAFFIC LIGHT CONTROLLER INTERFACE

5 weeks

4. 8086 Assembly Language Programming according to theory course using the following :

PC Assembler using TASM or MASM, TD or SYMDEB or CVD(Code View debugger)

2 weeks

CSE 227 **Operating Systems Laboratory** **CREDITS: 2**

INSTRUCTION: 3Periods/ Week **SESSIONAL MARKS: 50**
FINAL EXAM: 3Hrs **FINAL EXAM MARKS: 50**

PREREQUISITE: C Programming

COURSE OBJECTIVE:

1. To understand and write program in Unix environment
2. To design and implement the scheduling algorithms
3. To design and implement advanced file system operations

COURSE OUTCOMES:

1. Ability to write system level programs
2. Ability to develop shell scripts
3. Ability to understand the operating system .

Mapping of COs and POs

	POA	POB	POC	POD	POE	POF	POG	POH	POI	POJ	POK
CO1	2	--	3	-	3	-	-	3	2	3	2
CO2	3	-	3	-	2	-	-	3	2	3	-
CO3	2	2	-	-	3	-	-	-	3	-	-

LIST OF SAMPLE PROGRAMS

Write a C program for the following

1. Study of laboratory environment:Hardware specifications, software specifications
2. Simple Unix-C programs:Programs using system calls, library function calls to display and write strings on standard output device and files.
3. Programs using fork system calls.
4. Programs for error reporting using errno, perror( ) other system functions.
5. Programs using pipes.
6. Shell programming.Simple logic programs
7. C Programs to implement the shell commands
8. Programs to simulate process scheduling like FCFS, Shortest Job First and Round Robin.
9. Programs to simulate page replacement algorithms like FIFO, Optimal and LRU.
10. Programs to simulate free space management.
11. Programs to simulate virtual memory.
12. Program on deadlock management.
13. Programs to simulate deadlock detection.

REFERENCE BOOKS

1. Sumitabha Das, “Unix concepts and applications” ,TMH Publications.
2. Stevens, “Unix programming” , Pearson Education.
3. Yashwanth Kanetkar ,“Shell programming” .
4. Silberschatz, and Peter Galvin ,“Operating System Concepts” .

CSE 228 **Hardware Lab** **CREDITS: 2**

INSTRUCTION: 3Periods/ Week **SESSIONAL MARKS: 50**
FINAL EXAM: 3Hrs **FINAL EXAM MARKS: 50**

Prerequisite

Minimum computer hardware knowledge

Course Objectives:

1. To understand the need of PC hardware
2. To be able to install different operating systems
3. To be able to troubleshoot hardware problems
4. To be able to troubleshoot software problems

Course Outcomes:

1. Understanding of different computer peripherals and interfaces
2. Describe the architecture of various computer hardware devices and their functioning
3. Configure and install the different operating systems

	PO-A	PO-B	PO-C	PO-D	PO-E	PO-F	PO-G	PO-H	PO- I	PO-J	PO-K
CO-1	0	2	0	0	0	0	2	2	1	3	2
CO-2	1	2	1	3	1	0	2	3	1	0	0
CO-3	1	3	1	0	2	0	3	2	1	0	1

Week 1 & 2

The Instructor should explain the students about the PC Hardware like Motherboard, Processor, RAM, Hard Disk, Network Interface Card and other peripheral devices.

Week 2 & 3

Identify the peripherals of a computer, components in a CPU and its functions. Draw the block diagram of the CPU along with the configuration of each peripheral and submit to your instructor.

Week 4 & 5

Every student should disassemble and assemble the PC back to working condition. Lab instructors should verify the work and follow it up with a Viva.

Week 6 & 7

Every student should individually install operating system (Windows) in
the personal computer. Lab instructor should verify the installation and follow it up with a Viva.

Week 8 & 9

Every Student should individually install Operating system (Linux – Ubuntu) in the personal computer. Lab instructor should verify the installation and follow it.

Week 10 & 11

Every Student should individually install Operating system (Linux – Ubuntu) and Windows as **Dual Boot** in the personal computer. Lab instructor should verify the installation and follow it.

Week 12 & 13

Hardware Troubleshooting

Students have to be given a PC which does not boot due to improper assembly or defective peripherals. They should identify the problem and fix it to get
the computer back to working condition. The work done should be verified
by the instructor and followed up with a Viva.

Week 14 & 15

Software Troubleshooting

Students have to be given a malfunctioning CPU due to system software problems. They should identify the problem and fix it to get the computer back to working condition. The work done should be verified by the instructor and followed up with a Viva.

TEXT BOOKS

1. Peter Norton ,"Peter Norton's Inside the PC",. –, SAMS publications Eight Edition
2. Kate J. Chase ,"PC Hardware and A+ Handbook" , PHI (Microsoft)
3. Vikas Gupta, "Comdex Information Technology course tool kit" WILEY Dreamtech
4. Cheryl A Schmidt, "The Complete Computer upgrade and repair book", WILEY Dreamtech 3rd edition