Bharati Vidyapeeth

(Deemed to be University) Pune, India

College of Engineering, Pune

B.Tech. Computer Science & Engineering

(2023 Course)

Program Curriculum

As Per NEP Guidelines

VISION OF UNIVERSITY:

Social Transformation through Dynamic Education

MISSION OF UNIVERSITY:

- To make available quality education in different areas of knowledge to the students as per their choice and inclination.
- To offer education to the students in a conducive ambiance created by enriched infrastructure! and academic facilities in its campuses.
- To bring education within the reach of rural, tribal and girl students by providing them substantive fee concessions and subsidized hostel and mess facilities.
- To make available quality education to the students of rural, tribal and other deprived sections of the population

VISION OF THE INSTITUTE:

To be World Class Institute for Social Transformation Through Dynamic Education.

MISSION OF THE INSTITUTE:

- To provide quality technical education with advanced equipment, qualified faculty members, and infrastructure to meet needs of the profession and society.
- To provide an environment conducive to innovation, creativity, research, and entrepreneurial leadership.
- To practice and promote professional ethics, transparency and accountability for social community, economic and environmental conditions.

VISION OF THE DEPARTMENT

To be focused on innovative and quality education in computer science and engineering that prepares professionals for development of society.

MISSION OF THE DEPARTMENT

- To provide academic environment for the development of skilled professionals
- To cultivate research culture that contributes to the sustainable development of the society.
- To enhance academic and industry collaborations for global exposure.

PROGRAM EDUCATIONAL OBJECTIVES

The students of B.TECH. (Computer Science and Engineering), after graduating will able to,

1. Demonstrate technical and professional competencies by applying Engineering Fundamentals, knowledge of computing and technologies.
2. Exhibit effective personality, good communication and team building skills
3. Adopt to the latest trends in the field of computer science and engineering.

PROGRAM SPECIFIC OUTCOMES

1. To design, develop and implement computer programs on hardware towards solving problems.
2. To employ expertise and ethical practice through continuing intellectual growth and adapting to the working environment.

PROGRAM OUTCOMES

1 Apply the knowledge of mathematics, science, engineering, and computing to provide a solution of complex engineering problems.

2 Identify, analyse complex engineering problems to derive conclusions using computer science and engineering knowledge.

3 Outline resolutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration, societal, and environmental considerations.

4 Use existing research knowledge and research techniques including design of experiments, data analysis, and synthesis to provide valid inferences.

5 Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools.

6 Apply inferences obtained by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the subsequent responsibilities relevant to the professional engineering practice.

7 Recognize the impact of the professional engineering solutions in societal and environmental contexts to demonstrate the knowledge for sustainable development.

8 Apply ethical principles and execute professional ethics and responsibilities and norms of the engineering practice.

9 Work effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary surroundings.

10 Talk effectively on complex engineering activities with the engineering community such as being able to comprehend and write effective reports and design documentation, make effective presentations.

11 Prove knowledge and understanding of the engineering and management principles and apply these to one's work, as a member and leader in a team.

12 Recognise the need for and have the preparation and ability to engage in independent and life-long learning in context of technological change.

A. DEFINITION OF CREDITS:

1 Hour Lecture (L) per week	1 credit
1 Hour Tutorial (T) per week	1 credit
1 Hour Practical (P) per week	0.5 credits
2 Hours Practical (Lab)/week	1 credit

B. STRUCTURE OF UNDERGRADUATE ENGINEERING PROGRAMME:

Sr.No.	Category	Breakup of Credits
1	Basic Science Courses	16
2	Engineering Science Course	13
2	Core Courses and Lab	99
4	Professional Elective Courses	17
5	Project	09
6	Internship	06
7	Skill based Courses	16
**8	Value based Courses	08(Optional Credit)
9	Humanity/Social	06
TOTL		180

- **** Indicates optional credits**

C. COURSE CODE AND DEFINITION

Course Code	Definitions
L	Lecture
T	Tutorial
P	Practical
TW	Term Work
O	Oral
SEE	Semester End Examination
MJ	Major (Core) Courses
MI	Minor Courses
GE	General Elective Courses
OE	Open Elective Courses
SE	Skill Enhancement Courses
AE	Ability Enhancement Courses
VE	Vocational Enhancement Courses
VS	Vocational Skill Courses
VA	Value Added Courses
CC	Co-curricular Courses
EC	Extra-Curricular Courses
ID	Inter-disciplinary Courses
MD	Multidisciplinary Courses
RP	Research I Project Courses
PC	Practical Courses
ES	Engineering Science
AC	Audit Course

EC	Extracurricular Activities
BM	Basic Mathematics
BP	Basic Physics
BC	Basic Chemistry
UH	Universal Human Values

Semester wise Credits

Sr. No.	Semester	Credits
1	I	25
2	II	25
3	III	20
4	IV	20
5	V	23
6	VI	22
7	VII	23
8	VIII	22

BHARATI VIDYAPEETH (DEEMED TO BE UNIVERSITY)
COLLEGE OF ENGINEERING, PUNE
B. Tech. (CSE): Semester – III (NEP 2020 COURSE)

Sr. No	Category	Subject Code	Subject	Teaching Scheme			Examination Scheme-Marks							Credits		
				L	P	T	ESE	Internal Assessment	TW	PR	OR	Total	L	Pr/Or	T	Total
1.	MJ	MJ1104301	Database Systems	3	2	-	60	40	25	25	-	150	3	1	0	4
2.	MJ	MJ1104302	Non-Linear Data Structures	3	2	-	60	40	25	25	-	150	3	1	0	4
3.	MJ	MJ1104303	Software Engineering	3	-	1	60	40	25	-	-	125	3	-	1	4
4	MJ	MJ1104304	Machine Organization and Microprocessors	3	-	-	60	40	-	-	-	100	3	-	0	3
5.	MJ	MJ1104305	Object Oriented Methodology	3	2	-	60	40	25	-	-	125	3	1	-	4
6.	SE	SE1104306	Skill Base Course-III (Computer Skill Lab- I) Web Programming	-	2	-	-	-	25	25	-	50	0	1	0	1
			Total	**15**	**8**	**1**	**300**	**200**	**125**	**75**	**-**	**700**	**15**	**4**	**1**	**20**
7.	Audit Course-I	AC1113307	Indian Knowledge System	2	-	-	-	100	-	-	-	-	-	-	-	2
8.	*Value Added Course	VA1104308	VAC- I	2	-	-	-	100	-	-	-	-	-	-	-	2

***Indicate this is mandatory but the credits will not be considered in SGPA/CGPA.(As and when the students complete the course and submit the certificate, it should be reflected in the mark sheet. The student should clear the subject up to 7th Sem of his/her coursework.)**

BHARATI VIDYAPEETH (DEEMED TO BE UNIVERSITY)
COLLEGE OF ENGINEERING, PUNE
C. Tech. (CSE): Semester – IV (NEP 2020 COURSE)

Sr. No	Category	Subject Code	Subject	Teaching Scheme			Examination Scheme-Marks						Credits			
				L	P	T	ESE	Internal Assessment	TW	PR	OR	Total	L	P/O	T	Total
1.	MJ	MJ1104401	Theory of Computing	3	-	1	60	40	25	-	-	125	3	0	1	4
2.	MJ	MJ1104402	System Programming and Operating Systems	3	2		60	40	25		25	150	3	1	-	4
3.	MJ	MJ1104403	Computer Organization and Architecture	3	-	-	60	40	-	-	-	100	3	0	0	3
4.	MJ	MJ1104404	Computer Graphics and Multimedia	3	2	-	60	40	25	-		125	3	1	0	4
5.	MJ	MJ1104405	Computer Networks	3	2	-	60	40	25	25		150	3	1	-	4
6.	SE	SE1104406	Skill Base Course-IV (Python)	-	2	-	-	-	25	25		50	0	1	0	1
			Total	**15**	**8**	**1**	**300**	**200**	**125**	**75**	**25**	**700**	**15**	**4**	**1**	**20**
7.	*MOOC/ Swayam NPTEL	AE1104407	MOOC-I	-	-	-	-	-	-	-	-	-	-	-	-	2
8	*Social Activity	EC1104408	Social Activity	-	-	-	-	-	-	-	-	-	-	-	-	2

*** Indicate this is mandatory but the credits will not be considered in SGPA/CGPA.(As and when the students complete the course and submit the certificate, it should bereflected in the mark sheet. The student should clear the subject up to 7th Sem of his/her coursework.)**

B.Tech (Computer Science & Engineering) Semester- III

Database Systems

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	**03 Hours/Week**	**End Semester Examination** :	**60 Marks**	**Lecture:**	**03**
Practical:	**02 Hours/Week**	**Internal Assessment:**	**40 Marks**	**Practical:**	**01**
		Term Work:	**25 Marks**		
		Practical:	**25 Marks**		
		Total:	**150 Marks**	**Total:**	**04**

Course Objective:

1. The Objective Of The Course Is To Present An Introduction To Database Management System As A Subject In Its Own Right.
2. To Understand The Fundamental Concepts Of Relational Database Management System..
3. To Provide A Strong Formal Foundation In Relational Database Concepts, Database Concepts Technology And Practice &To Introduce The Concepts Of Query Processing.
4. To Present SQL And Procedural Interfaces To SQL Comprehensively.
5. To Introduce The Concepts Of Transaction Processing And To Present The Issues And Techniques Relating To Concurrency And Recovery In Multi-User Database Environments.
6. To Introduce The Recent Trends In Database Technology.

Prerequisite:

Discrete Mathematics, Data Structures And Programming Languages.

Course Outcomes: On Completion Of The Course, Students Will Have The Ability To:

1. Understand Fundamental Elements Of Database Management Systems.
2. Describe The Fundamental Elements Of Relational Database Management Systems And Design ER Models To Represent Simple Database Application Scenarios.
3. Apply Normalization On Database Design To Eliminate Anomalies.
4. Create SQL Queries To Interact With Database.
5. Analyze Database Transactions And Can Control Them By Applying ACID Properties.
6. Apply Recent Trends In Database Technology.

Unit I : Introduction To Databases **06 Hours**

Basic Concepts, Advantages Of DBMS Over File Processing Systems, Data Abstraction, Database Languages, Data Models, Data Independence, Components Of A DBMS, Overall Structure Of DBMS, Multi-User DBMS Architecture, System Catalogs, Data Modeling: Basic Concepts, Entity, Attributes, Relationships, Constraints, Keys.

Unit II: Relational Model **06 Hours**

Components Of ER Model, Conventions, Converting ER Diagrams Into Tables Relational Model: Basic Concepts, Attributes And Domains, Codd's Rules. Relational Integrity: Nulls, Entity, Referential Integrities, Enterprise Constraints, Views, Schema Diagram.

Unit III: Relational Databases Design **06 Hours**

Purpose Of Normalization, Data Redundancy And Update Anomalies, Functional Dependencies. The Process Of Normalization: 1NF, 2NF, 3NF, BCNF. Introduction To Query Processing: Overview, Measures Of Query Cost, Selection And Join Operations, Evaluation Of Expressions Introduction To Query Optimization: Estimation, Transformation Of Relational Expression, Triggers.

Unit IV: Introduction To SQL **06 Hours**

Characteristics And Advantages SQL Data Types, Literals, DDL, DML, SQL Operators Tables: Creating, Modifying, Deleting, Views: Creating, Dropping, Updation Using Views, Indexes, Nulls SQL DML Queries: SELECT Query And Clauses, Set Operations, Tuple Variables, Set Comparison, Ordering Of Tuples , Aggregate Functions, Nested Queries, Database Modification Using SQL Insert, Update, Delete Queries, Stored Procedure.

Unit V: Transaction Management **06 Hours**

Basic Concept Of A Transaction, Properties Of Transactions, Database Architecture, Concept Of Schedule, Serial Schedule. Serializability: Conflict And View, Cascaded Aborts Recoverable And Non-Recoverable Schedules. Concurrency Control: Need Locking Methods Dead Locks, Timestamping Methods. Optimistic Techniques, Multi-Version Concurrency Control. Different Crash Recovery Methods: Shadow-Paging, Log-Based Recovery: Deferred And Immediate, Check Points.

Unit VI: Database Architectures **06 Hours**

Centralized And Client-Server Architectures, 2 Tier And 3 Tier Architecture, Introduction To Parallel Databases, Key Elements Of Parallel Database Processing, Architecture Of Parallel Databases, Introduction To Distributed Databases, Architecture Of Distributed Databases, Distributed Database Design.
Emerging Database Technologies: Introduction, No SQL Databases- Internet Databases, Cloud Databases, Mobile Databases, XML Databases.

Textbooks:

1. Silberschatz A., Korth H., Sudarshan S. “Database System Concepts”, 6th Edition, Tata Mcgraw Hill Publishers
2. G. K. Gupta “Database Management Systems” , Tata Mcgraw Hill

Reference Books:

1. Rab P., Coronel C. “Database Systems Design, Implementation And Management”, 5th Edition, Thomson Course Technology, 2002
2. Elmasri R., Navathe S. “ Fundamentals Of Database Systems”, 4th Edition, Pearson Education, 2003
3. Date C. “ An Introduction To Database Systems”, 7th Edition, Pearson Education, 2002
4. Ramkrishna R., Gehrke J. “ Database Management Systems”, 3rd Edition, Mcgraw Hill

List Of Assignments:

1. Define Database. Explain Importance Of Data Models In Detail
2. Write A Short Note On Following:

A) Relational Algebra And Calculus

B) Integrity Constraints

3. Define Normalization. Explain All Normal Forms In Detail
4. Explain DDL, DML And DCL In Detail.
5. Explain ACID Properties Of TCL In Detail.
6. Explain NOSQL And Database Analysis Tools In Detail.

List Of Laboratory Exercises:

1. Draw E-R Diagram And Extended ER Diagram (As Given By Course Coordinator) On Given System.
2. Demonstrate Queries On Relational Algebra(As Given By Course Coordinator).
3. To Discuss Normalization And Build Normalized Schema (As Given By Course Coordinator)On Given System.
4. Write A SQL Statement ((As Given By Course Coordinator) On DDL,DML And DCL.
5. Demonstrate Queries On Joins(As Given By Course Coordinator).
6. Demonstrate Queries On Aggregate Functions(As Given By Course Coordinator).

7. Use WEKA Tool To Derive Analytical Model For The Given Dataset.
8. Case Study On NOSQL Database: Mongodb.

Project Based Learning:
1. Inventory Management System.
2. Online Jewelry Shopping System
3. Library Management System
4. Online Examination System
5. Hospital Management System
6. Railway Reservation System
7. Payroll Management System
8. Cooking Recipe Portal
9. Art Gallery Management System
10. Student Database Management System
11. Restaurant Management Database System
12. Electric Bill System Database
13. Online Examination System
14. Event Management System
15.Attendance Management System

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Non-Linear Data Structures

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Theory:	03 Hours/Week	End Semester Examination :	60 Marks	Theory	03
Practical:	02 Hours/Week	Internal Assessment:	40 Marks		
		Term work:	25 Marks	Practical	01
		Practical:	25 Marks		
		Total:	150 Marks	Total	04

Course Objective:
The Objective Of The Course Is To Provide The Students The Knowledge Of Different Non-Linear Data Structures And How To Use These To Solve Real World Problems.

Prerequisite:
Basic Knowledge Of Algorithm, Programming Fundamentals, Data Types, ADT, Linear Data Structure.

Course Outcomes: On Completion Of The Course, Students Will Have The Ability To:

1. Identify And Analyze Appropriate Algorithms On The Graph To Solve Real World Problems.
2. Demonstrate The Use Of Trees And Binary Search Trees To Solve The Real World Problem.
3. Compare The Different Types Of Trees Like AVL Trees, Btree, B+ Tree, Red Black Tree And Select An Appropriate One To Solve A Particular One
4. Perform Insertion And Deletion Operation On Heap.
5. Apply Appropriate Hash Function For A Search Process.
6. Implement The Concept Of Dictionaries And Text Processing.

Unit I: Graphs **06 Hours**
Introduction To Non-Linear Data Structure, Graphs, Representation Of Graph, AND/OR Graphs, ADT For Graph, Traversing A Graph, Dijkstra's Algorithm, Minimum Spanning Trees.

Unit II: Trees **06 Hours**
Introduction, Binary Trees, Binary Tree Representation, Tree Traversal Algorithms, Threaded Binary Tree, Binary Search Tree, Operations On Binary Search Tree, Huffman's Algorithm.

Unit III: Special Forms Of Trees **06 Hours**
AVL Trees, M-Way Search Trees, B Trees, B+ Trees, Red Black Tree, 2-3 Trees, Splay Trees, Applications Of Trees.

Unit IV: Heaps **06 Hours**
Heaps As Priority Queues, Heap Implementation, Insertion And Deletion Operations, Binary Heaps, Binomial And Fibonacci Heaps, Heapsort, Heaps In Huffman Coding.

Unit V: Hashing **06 Hours**
Introduction, Hash Functions, Collision Resolution Strategies,
Types Of Hashing, Separate Chaining, Open Addressing, Linear Probing, Quadratic Probing, Double Hashing, Rehashing, Extendible Hashing.

Unit VI: Dictionaries & Text Processing **06 Hours**
Definition, Dictionary Abstract Data Type, Implementation Of Dictionaries, Text

Processing: String -+Operations, Brute-Force Pattern Matching, The Boyer-Moore Algorithm, The Knuth-Morris-Pratt Algorithm, Standard Tries, Suffix Tries, The Huffman Coding Algorithm.

Textbooks:

1. Mark Allen Weiss, Data Structures And Algorithm Analysis In C++, 2nd Edition, Pearson, 2004.
2. M T Goodrich, Roberto Tamassia, Algorithm Design, John Wiley, 2002.
3. Fundamentals Of Data Structures In C By Horowitz, Sahni& Anderson-Freed, 2e Universal Press

Reference Books:

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest, And C. Stein, Introduction To Algorithms, MIT Press, 3/E, 2009.
2.A. M. Tenenbaum, Y. Langsam, And M. J. Augenstein, Data Structures Using C And C++,Prentice Hall, 3/E.

List Of Assignments:

The Following Are Some Sample Assignments. The Course Co-Ordinator Will Frame One Assignment On Each Unit For Internal Assessment.

1. Apply The Shortest Path Algorithm On The Given Graph.
2. Apply The Appropriate Algorithm And Find The Solution For The Problem.
3. Generate Huffman Code
4. Write A Pseudocode For Tree Traversal Operation
5. Explain The Concept Of Linear Probing
6. Explain The Steps In Text Processing

Project Based Learning

1. Hashing For Cryptography
2. Payroll System
3. Network Route Identifier
4. Path Finder
5. Telephone Directory
6. Library Management System
7. Document Indexing
8. Data Compressor
9. Railway Reservation System
10. Supermarket Stock Management

List Of Laboratory Exercises:

1. Apply Graph Traversal Technique.
2. Demonstrate Use Of Dijkstra's Algorithm.
3. Perform Operations On Binary Search Trees.
4 Perform On Tree Traversal Algorithms.
5. Applications And Demonstration On Different Types Of Trees.
6. Perform The Operations On Heaps.
7. Apply Hash Function To Solve The Real Time Problem.
8. Demonstrate Use Of Dictionaries And Concept Of Text Processing.
9. Mini Project

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Software Engineering

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	03 Hours/Week	End Semester Examination :	60 Marks	Lecture	03
Tutorial:	01 Hour/Week	Internal Assessment:	40 Marks		01
		Term work:	25 Marks		
		Total	125Marks	Total	04

Course Objective:

The Main Purpose Of This Course Is To Impart Knowledge On The Basic Principles Of The Software Development Life Cycle.

Prerequisite: Programming Paradigms, Basic Mathematical Ability

Course Outcomes: On Completion Of The Course, Students Will Have The Ability To:

1. Understand The Basic Concepts Of Software Engineering Lifecycle.
2. Analyze And Categorize The Requirements Of Software Systems.
3. Design The Software Qualitatively.
4. Implement The Software With The Standard Guidelines.
5. Validate The Software With Standard Testing Techniques.
6. Implement Troubleshooting Of Software Application.

Unit I: Introduction **06 Hours**

Software And Software Engineering: The Nature Of Software, The Software Process, Software Myths, Process Models: A Generic Process Model, Prescriptive Process Models, Specialized Process Models, The Unified Process, Agile Development: Agility, Agility And The Cost Of Change, Agile Process, Agile Process Models.

Unit II: Requirement Engineering And Modelling **06 Hours**

Understanding Requirements: Requirements Engineering, Eliciting Requirements, Building The Use Case, Building Analysis Model, Negotiating Requirements And Validating Requirements. Drafting The Software Requirement Specification.

Requirement Analysis And Modelling: Domain Analysis, Object Oriented Analysis, Scenario Based Modelling, Class Based Modelling, Behavior Modelling. CASE Tools.

Unit III: Software Project Management **06 Hours**

Introduction To Software Project Management, Selection Of A Project Approach, Project Estimation Techniques, Project Planning And Project Scheduling, Project Organization And Team Structures, Risk Management, Resource Allocation, Project Monitoring And Control, Software Configuration Management, Software Quality Management, CASE Tool.

Unit IV: Design **06 Hours**

Design Concepts: The Design Process, Design Model - Data Design Model, Architecture Design Model, Transform And Transaction Flow, Interface Design Flow, Component Level And Deployment Level Design Elements.

Design Concepts – Abstraction, Architecture, Patterns, Modularity, Functional Independence, Refinement, Refactoring, Object-Oriented Design Concepts. CASE Tools In Software Design.

Unit V: Coding And Testing **06 Hours**

Coding Approach, Coding Standards, Error, Bug, Defects.

Software Testing Life Cycle, Software Testing Principles, Verification And Validation, Types Of Testing, White Box Testing Techniques, Black Box Testing

Techniques, Testing Oo Applications, Website Testing. Case Tool.

06 Hours

Unit VI: Implementation And Maintenance

Software Maintenance-Software Supportability.
Reengineering-Business Process Reengineering- Software Reengineering- Reverse Engineering, Restructuring, Forward Engineering- Economics Of Reengineering.

Textbooks:

1. Roger S, "Software Engineering – A Practitioner's Approach", Seventh Edition, Pressman, 2010.
2. Pearson Edu, "Software Engineering By Ian Sommerville", 9th Edition, 2010.

Reference Books:

1. Van Vliet, "Software Engineering: Principles And Practices"–, 2008.
2. Richard Fairley, "Software Engineering Concepts", 2008..

List Of Assignments:

1. Presentation On One Topic Related To This Syllabus
2. Conducting Six Multiple Choice Question Online Test On Each Unit
3. Prepare A Report Based On the Understanding by Viewing The NPTEL Videos Of This Subject.
   i. Perform The Following Assignments:
   ii. Suggest Which SDLC Model Will Be Used to Develop ATM Software? Justify?
   iii. Develop Requirement Specification for an ATM Software.
   iv. Design The Class Diagram and Use Case Diagram for ATM Software.
   v. Discuss On Automated Software Testing. Create Test Cases for Functionality of ATM Software Using a Test Tool (Test Link).
   vi. Discuss and Prepare IEEE Quality Document for ATM Software.
   vii. Prepare Gantt Chart Using MS Project CASE Tool For A Small Project.

Project Based Learning:

1. ATM System
2. Online Banking System
3. Airline Reservation System
4. Railway Reservation System
5. Library Management System
6. College Management System
7. Hospital Management System
8. Traffic Monitoring System
9. Hotel Management System
10. Bus Ticket Reservation System
11. Online Shopping System

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Machine Organization and Microprocessor

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Theory:	03 Hours/Week	End Semester Examination	60 Marks	Theory:	03
		Internal Assessment:	40 Marks		
		Total:	100 Marks	Total:	03

Course Objective:
The Course Would Provide Students with an Understanding of the Architecture and Programmer's Model of Advanced Processor and Provide Practical Exposure On Microprocessor. Students Will Learn The Basic Operations And The System Level Features Of Advanced Processor To Apply The Assembly Language Programming To Develop Small Real Life Embedded Application.

Prerequisite:
The Students Should Have Basic Knowledge Digital Electronics And Logic Design.

Course Outcomes: On Completion Of The Course, Students Will Have The Ability To:
1. Understand Instruction Set And Assembly Language Programming.
2. Remember The Architecture And Features Of Microprocessor.
3. Analyze Different Processor Modes.
4. Apply Exceptions And Interrupt Mechanism In Application.
5. Identify And Analyze The Tools Used To Design And Debug Microprocessor Based Systems.
6. Create Systems Using Microprocessor And Microcontroller For Real Time Applications.

Unit I: Introduction to 80386 Architecture **06 Hours**

Brief History of Intel Processors, 80386 Features and Architecture, Operating Modes, Addressing Modes, Data Types.
Applications Instruction Set- Data Movement Instructions, Binary Arithmetic Instructions, Decimal Arithmetic Instructions, Logical Instructions, Control Transfer Instructions, String And Character Transfer Instructions .

Unit II: Systems Architecture and Memory Management **06 Hours**

System Registers (Flag Registers, Memory Management Registers, Control Registers, Debug Register, Test Registers), System Instructions.
Memory Management- Global Descriptor Table, Local Descriptor Table, Interrupt Descriptor Table, Segment Translation, Page Translation, Combining Segment And Page Translation.

Unit III: Protection and Multitasking **06 Hours**

Protection- Need of Protection, Overview of 80386DX Protection Mechanisms, Segment Level Protection, Page Level Protection, Combining Segment and Page Level Protection. Multitasking- Task State Segment, TSS Descriptor, Task Register, Task Gate Descriptor, Task Switching, Task Linking, Task Address Space.

Unit IV: Input-Output, Exceptions And Interrupts **06 Hours**

Input-Output- I/O Addressing, I/O Instructions, Protection and I/O Exceptions And Interrupts- Identifying Interrupts, Enabling And Disabling Interrupts, Priority Among Simultaneous Interrupts And Exceptions, Interrupt Descriptor Table (IDT), IDT Descriptors, Interrupt Tasks And Interrupt Procedures, Error Code, And Exception Conditions.

Unit V: Initialization of 80386, Debugging **06 Hours**

Initialization- Processor State After Reset, Software Initialization for Real

Address Mode, Switching to Protected Mode, Software Initialization For Protected Mode, Initialization Example, TLB Testing Debugging- Debugging Features Of The Architecture, Debug Registers, Debug Exceptions, Breakpoint Exception.

Unit VI: Virtual 8086 Mode and Introduction To Microcontrollers **06 Hours**

Virtual 8086 Mode- Executing 8086 Code, Structure of V86 Stack, Entering and Leaving Virtual 8086 Mode. Architecture Of Typical Microcontroller, Difference Between Microprocessor And Microcontroller, Characteristics Of 8 Bit And 16 Bit Microcontrollers, Application Of Microcontrollers.

Textbooks:

1. Douglas Hall, "Microprocessors & Interfacing", Mcgraw Hill, Revised Second Edition, 2006 ISBN 0-07-100462-9
2. A.Ray, K.Bhurchandi, "Advanced Microprocessors And Peripherals: Arch, Programming & Interfacing", Tata Mcgraw Hill,2004 ISBN 0-07-463841-6
3. James Turley, "Advanced 80386 Programming Techniques", Mcgraw-Hill, ISBN: 10: 0078813425, 13: 978-0078813429.
4. Intel 80386 Programmer's Reference Manual 1986, Intel Corporation, Order No.: 231630-011, December 1995.
5. Intel 80386 Hardware Reference Manual 1986, Intel Corporation, Order No.: 231732-001, 1986

Reference Books:

1. Chris H. Pappas, William H. Murray, ―80386 Microprocessor Handbooks‖, Mcgraw-Hill Osborne Media, ISBN-10: 0078812429, 13: 978-0078812422..
2. Walter A. Triebel, ―The 80386Dx Microprocessor: Hardware‖, Software, And Interfacing, Pearson Education, ISBN: 0137877307, 9780137877300.
3. Brey, Barry B, ―8086/8088, 80286, 80386 And 80486 Assembly Language Programming‖, Prentice Hall, ISBN: 13: 9780023142475.
4. Introduction To 64 Bit Intel Assembly Language Programming For Linux, 2nd Edition, Ray Seyfarth, ISBN10: 1478119209, ISBN-13: 9781478119203, 2012.
5. Mohammad Rafiquzzaman, ―Microprocessors: Theory And Applications: Intel And Motorola", Prentice Hall, ISBN:-10:0966498011, 13:978:0966498011.
6. Assembly Language Step-By-Step: Programming With Linux, 3rd Edition, Jeff Duntemann, Wiley ISBN:-10 0470497025, ISBN-13: 978-0470497029, 2009.

Project Based Learning

1. Auto Controlled Lights.
2. Gate Controller, Automated
3. A Basic 16-Bit Calculator
4. Temperature Sensor
5. Create A Text-Based Adventure Game with Multiple Choices and Outcomes.
6. Build A Memory Game Where the Player Has to Match Pairs of Numbers or Symbols.
7. Design A Digital Clock That Displays Hours, Minutes, And Seconds.
8. Timing Events with A Timer/Counter and an Interrupt Service Routine
9. Number Converter.

List Of Laboratory Exercises:

1. Write X86/64 ALP To Count Number Of Positive And Negative Numbers From The Array.
2. Write An ALP To Accept Five 64 Bit Hexadecimal Numbers From User And Store Them In An Array And Display The Accepted Numbers.
3. Write X86/64 ALP To Perform Non-Overlapped Block Transfer (With And Without String Specific Instructions). Block Containing Data Can Be Defined In The Data Segment.
4. Write X86/64 ALP To Perform Overlapped Block Transfer (With And Without String Specific

Instructions). Block Containing Data Can Be Defined In The Data Segment.

5. Write X86/64 ALP To Perform Multiplication Of Two 8-Bit Hexadecimal Numbers. Use Successive Addition And Add And Shift Method. (Use Of 64-Bit Registers Is Expected).
6. Write X86/64 ALP To Convert 4-Digit Hex Number Into Its Equivalent BCD Number And 5-Digit BCD Number Into Its Equivalent HEX Number. Make Your Program User Friendly To Accept The Choice From User For: (A) HEX To BCD B) BCD To HEX (C) EXIT. Display Proper Strings To Prompt The User While Accepting The Input And Displaying The Result. (Wherever Necessary, Use 64-Bit Registers).
7. Write X86 Assembly Language Program (ALP) To Implement The Following OS Commands I. TYPE Ii. COPY And Iii. DELETE Using File Operations. Users Are Supposed To Provide Command Line Arguments In All Cases.
8. Write X86 ALP To Find, A) Number Of Blank Spaces B) Number Of Lines C) Occurrence Of A Particular Character. Accept The Data From The Text File. The Text File Has To Be Accessed During Program_1 Execution And Write FAR PROCEDURES In Program_2 For The Rest Of The Processing. Use Of PUBLIC And EXTERN Directives Is Mandatory.
9. Write X86 Program To Sort The List Of Integers In Ascending/Descending Order. Read The Input From The Text File And Write The Sorted Data Back To The Same Text File Using Bubble Sort.
10. Write X86/64 ALP To Switch From Real Mode To Protected Mode And Display The Values Of GDTR, LDTR, IDTR, TR And MSW Registers Also Identify CPU Type Using CPUID Instruction.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Object Oriented Methodology

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Theory:	03 Hours/Week	End Semester Examination:	60 Marks	Theory :	03
Practical:	02 Hours/Week	Internal Assessment	40 Marks	Practical:	01
		Term Work	25 Marks		
		Total	125 Marks	Total :	04

Course Objective:

1. To Teach Fundamentals Of Object-Oriented Concepts And Programming.
2. To Apply The Concepts Of Object-Oriented Paradigm.
3. To Develop Object-Oriented Programming Skills.
4. To Design And Implement Applications For Real Life Problems By Using Object-Oriented Programming.

Prerequisite:

Paradigms of Programming.

Course Outcomes: On Completion Of The Course, Students Will Have The Ability To:

1. Understand The Basics Of Object-Oriented Programming With Java.
2. Learn The Concept Of Class, Objects, Methods For Real Time Problems.
3. Analyze Constructor, Garbage Collector And Methods Of String Class.
4. Explore The Concept Of Inheritance And Polymorphism With The Help Of Real Time Applications.
5. Apply The Concept Of Exception Handling And Understand The Concept Of Multithreading.
6. Design The Graphical User Interface By Using Applets AWT And SWING.

Unit I: Introduction **06 Hours**

Overview Of Oop, Object Oriented Programming Paradigms, Features Of Object Oriented Programming, Difference Between Java, C And C++, Structure Of Java Program, Difference Between Jdk, Jre And Jvm, Internal Details Of Jvm,
Basics Of Java Language-Variable, Arrays And Reserve / Keywords Present In Java, Primitive Data Types, Java Operators, Decision Making And Branching Statements In Java.

Unit II: Classes, Objects And Methods **06 Hours**

Creating A Class, Visibility/Access Modifiers, Encapsulation, Methods: Adding A Method To Class, Returning A Value, Adding A Method With Parameters,' This' Keyword, Method Overloading, Object Creation, Using Object As A Parameters, Returning Object, Array Of Objects, Memory Allocation: 'New', Static Data Members, Static Methods.
Packages- Defining A Package, Classpath, Importing Packages
Stream Based I/O (Java.Io) – The Stream Classes-Byte Streams And Character Streams, Reading Console Input And Writing Console Output, Enumerations, Autoboxing, Generics.

Unit III: Constructors, Destructors And String Handling: **06 Hours**

Use Of Constructor, Characteristics Of Constructors, Types Of Constructor, Constructor Overloading, Constructor With Default Arguments, Symbolic Constants, Garbage Collection, Destructors And Finalizers.

String Handling: Immutable String, String Comparison, String Concatenation, Substring, Methods Of String Class, String Buffer Class, Stringbuilder Class, Creating Immutable Class, To String Method.

06 Hours

Unit IV: Inheritance And Polymorphism

Use Of Inheritance, Types Of Inheritance In Java, Role Of Constructors In Inheritance, Polymorphism In OOP, Types Of Polymorphism, Static And Dynamic Polymorphism, Overriding Super Class Methods. Use Of "Super" Keyword. Interfaces, Implementing Interfaces.

Interfaces- Defining An Interface, Implementing Interfaces, Nested Interfaces, Applying Interfaces, Variables In Interfaces And Extending Interfaces.

Unit V: Exception Handling And Multithreaded Programming **06 Hours**

Exception Handling: Try And Catch Block, Catch Block, Nested Try, Finally Block, Throw Keyword, Exception Propagation, Throws Keyword, Exception Handling With Method Overriding, Custom Exception

Introduction To Threads: Life Cycle Of A Thread, Thread States, Thread Properties, Methods In Threads And Runnable, Setting Priority Of Threads, Synchronization And Inter Thread Communication Life Cycle Of A Thread.

Unit VI: Designing Graphical User Interfaces In Java **06 Hours**

Applet And Its Use, Design Patterns Using Applet And Japplet. Run Applet Application By Browser And Applet Tool. Applet Architecture. Parameters To Applet Life Cycle.

Basics Of Components Using Containers, Layout Managers And User Defined Layout. Border Layout, Flow Layout, Grid Layout, Grid Bag Layout, Box Layout. Awt Components, Adding A Menu To Window, Extending Gui Features Using Swing, Components Designing Gui, Advanced Swing Components Like Progress, Jslider, Jradio Button, Jtree, Jtable, Jtoggle Button, Etc.

Textbooks:

1. E. Balaguruswamy, "Object Oriented Programming Using C++ And Java", Tata Mcgrawhill
2. Steven Holzner Et Al. "Java 2 Programming", Black Book, Dreamtech Press, 2009.

Reference Books"

Java The Complete Reference, Herbert Schildt, Mcgraw Hill Education (India) Pvt. Ltd. 9th Edition, 2014, ISBN: 978-0-07-180856-9 (E-Book).

1. Object-Oriented Design Using Java, Dale Skrien, Mcgraw-Hill Publishing, 2008, Isbn - 0077423097, 9780077423094.
2. Mitsunori Ogihara, "Fundamentals Of Java Programming", Springer; 2018, Isbn 978-3-319-89490-4.
3. Brahma Dathan Sarnath Ramnath, "Object-Oriented Analysis, Design And Implementation An Integrated Approach", Springer; 2nd Ed. 2015, Issn 1863-7310 Issn 2197-1781 (Electronic) Undergraduate Topics In Computer Science Isbn 978-3-319-24278-1, Isbn 978-3-319-24280-4.
4. T. Budd (2009), An Introduction To Object Oriented Programming, 3rd Edition, Pearson Education, India.
5. J. Nino, F. A. Hosch (2002), An Introduction To Programming And Oo Design Using Java, John Wiley & Sons, New Jersey.
6. Y. Daniel Liang (2010), Introduction To Java Programming, 7th Edition, Pearson Education, India.

Project Based Learning:

1. Smart City Project
2. Currency Converter
3. Online Exam Project In Java
4. Moving Balls Mini Project Using Java Applet
5. Text Editor In Java Using Awt Controls.
6. Album Manager Project In Java
7. Vehicle Management System In Java
8. Music Player Project In Java
9. Student Management System Project In Java
10. Simple Calculator Project In Java
11. Image To Pdf Convertor In Java
12. Simple Chat System
13. Online Quiz Project
14. Pong Game In Java
15. Tokenize Implementation.

List Of Laboratory Exercises:

1. Write A Program That Checks Whether A Given String Is A Palindrome Or Not.
2. Write A Program That Describes A Class Person. It Should Have Instance Variables To Record Name, Age And Salary. Create A Person Object. Set And Display Its Instance Variables.
3. Write A Program That Creates A Class Circle With Instance Variables For The Centre And The Radius. Initialize And Display Its Variables.
4. Write A Program That Counts The Number Of Objects Created By Using Static Variable.
5. Write A Program To Demonstrate The Constructors In Java.
6. Write A Program To Demonstrate The Constructor Overloading.
7. Write A Program To Display The Use Of This Keyword.
8. Write A Program To Implement Class And Inheritance Concept.
9. Write An Application That Creates An Interface' And Implement It.
10. Write A Program That Can Count The Number Of Instances Created For The Class.
11. Write A Program To Implement The Concept Of Multithreaded Programming.
12. Create An Abstract Class Shape. Let Rectangle And Triangle Inherit This Shape Class. Add Necessary Functions.
13. Write An Application That Shows The Usage Of Try, Catch, Throws And Finally.
14. Write An Applet That Displays ―Hello World (Background Colour-Black, Text Colour-Blue And Your Name In The Status Window).
15. Develop Mini Project Using Applet And Swings.

Syllabus For Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

CSE Skill Lab – I (Web Programming)

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	0 Hrs	University Examination:	-		
Practical:	2 Hrs	Internal Assessment:	-	Lecture	-
		Term Work	25	Practical	1
		Practical	25		
		Total	50	Total	1

Course Objective: The Objective Of This Course Is To Impart Students With The Knowledge To Setup The Development Environment, Design And Develop Dynamic Database Driven Web Applications Using PHP.

Prerequisite:
Basics Of HTML

Course Outcomes: On Completion Of The Course, Students Will Have The Ability To:

1. Create PHP Programs That Use Various Php Library Functions,
2. Design Interactive Forms Using PHP.
3. Implement Various Operations On Arrays And Control Structures In PHP
4. Create Database And Demonstrate The Manipulation Of Files, Directories And Relational Data.
5. Implement Cookies And Session
6. Develop Dynamic Web Content.

Unit I : Introduction To Dynamic Web Content &Environment: **Hours**

Overview Of HTTP & HTML, Request/Response Procedure, Advantage Of PHP, MySQL, JavaScript, CSS & HTML 5, The Apache Web Server, Overview Of Open Source. Basics Of WAMP, MAMP, LAMP, Installation, Accessing Document Root, Working Remotely, Using FTP, Using IDE. **6 Hours**

Unit II : Introduction To PHP, Expression & Control Flow:: **6 Hours**

Incorporating PHP Within HTML, Structure Of PHP – Comments, Basic Syntax, Variables, Operators, Assignments, Multiline Commands, Constants, Echo & Print Commands, Functions, Variable Scope. Expressions: TRUE Or FALSE, Literals & Variables. Operators: Precedence, Associativity, Relational Operators. Conditionals – If, Else, Else if, Switch Operator. Looping: While, Do-While, Breaking Out Of Loop, Continue Statement

Unit III : PHP Functions, Objects &Array **6 Hours**

PHP Functions: Defining, Returning A Value, Returning An Array, Do Not Pass Arguments By Reference, Returning Global Variables, Include Statement: Include Once, Require &Require Once.
PHP Objects: Declaring A Class, Creating An Object, Accessing Objects, Constructors, PHP Destructors, Writing Methods, Declaring Properties & Constants, Inheritance.
Arrays: Basic Access, Foreach As Loop, Multidimensional Arrays, Using Array Functions.

Unit IV: PHP In Action & Introduction To MySQL **6 Hours**

Using printf, Date And Time Functions, File Handling, System Calls. Introduction To MySQL: Basics, Database Terms, Accessing MySQL Via Command Line, MySQL Commands, Data Types, Indexes, MySQL Functions, Accessing MySQL Via Phpmyadmin, Primary Keys, Relationships, Select Queries, Backup File, Dumping Data In CSV Format.

Unit V : Cookies, Sessions, Authentications And Accessing **6 Hours**

Using Cookies In PHP, HTTP Authentication, Using Sessions. Accessing: Querying A Mysql Database With PHP – The Process, Create Login File, Connecting To Database, Practical Example, Preventing Hacking Attempts: Using Placeholders, HTML Injection. Building Forms, Retrieving Submitted Data.

Unit VI: Exploring Javascript **6 Hours**

Javascript And HTML Text – Using Within A Document Head, Older & Nonstandard Browsers, Including Javascripts, Debugging Javascript Errors, Using Variables, Semicolon, Variables, Operators, Variable Typing, Functions, Global Variables, Local Variables, Document Object Model.

Textbooks:

1. Learning PHP, Mysql &Javascript, Robin Nixon, OREILLY, 4thediton, 2015.
2. Head First PHP & Mysql-Lynn Beighley& Michael Morrison-O'Reilly.
3. PHP: A Beginner's Guide-Vikramvaswani- Mcgraw-Hill Education.

Reference Books:

1. The Complete Reference PHP – Steven Holzner, Tata Mcgraw-Hill.
2. The Joy Of PHP Programming: A Beginner's Guide – Alan Forbes, Beakcheck LLC,6th Edition.

List Of Laboratory Exercises:

1. Write A PHP Function To Count Total Number Of Vowels From The String. Accept A String Using HTML Form.
2. Write A PHP Script To Print Fibonacci Series.
3. Create A Student Registration Form Using Text Box, Check Box, Radio Button, Select, Submit Button. Display The User Inserted Value In New PHP Page.
4. Write A Program To Perform The Following Operations On An Associative Array.
    a) Display Elements Of An Array Along With Their Keys.
    b) Display Size Of Array.
    c) Delete An Element From An Array From The Given Index.
5. Write A Program To Insert A Roll No And Student Name In A Database (Use Postgresql Data To Create
6. Write PHP Script To Demonstrate Passing Variables With Cookies.
7. Implement Admin Login/Logout Functionality And Cookie Wherever Required.
8. Write A PHP Script To Connect MYSQL Server From Your Web Application. Write A PHP Script To Create And Drop Database.
9. Create Database Using Phpmyadmin. Write A Program To Read Input Data From Table And Display The Information In Tabular Form.
10. Develop PHP Application Using Forms And Database.

Project Based Learning:

1. Design Personal Website Using Html And Css
2. Login Page With User Credentials And Data Base
3. Drawing Palette Based Web Page
4. Student Registration Form With Data Base Connectivity
5. Dummy Social Networking Website
6. Image Editing Using We Page
7. Animation Based Website8. Simple Game Website
9. Ecommerce Website With Data Base Connectivity
10. Any Complete Web Project With Real Time Database Connectivity
11. Login Authentication
12. Design Survey Form
13. Quiz Game

14. Implement Employee Management System
15. Social Media Dashboard
16. Search Application
17. E-Commerce Website
18. Develop Freelance Platform
19. Hospital Management System

B.Tech
(Computer Science & Engineering)Semester- IV

Theory of Computing

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Theory:	03 Hours/Week	End Semester Examination:	60 Marks	Theory:	03
Tutorial:	01 Hours/Week	Internal Assessment:	40 Marks	Tutorial:	01
		Term Work:	25 Marks		
		Total:	125 Marks	Total:	04

Course Objective:

This Course Introduces Some Fundamental Concepts In Automata Theory And Formal Languages Including Grammar, Finite Automaton, Regular Expression, Formal Language, Pushdown Automaton And Turing Machine. This Subject Not Only Forms The Basic Models Of Computation, It Also Includes The Foundation Of Many Branches Of Computer Science, E.G. Compilers, Software Engineering, Concurrent Systems, Etc.

Prerequisite:
Discrete Mathematics

Course Outcomes: On Completion Of The Course, Students Will Have The Ability To:

1. Demonstrate Advanced Knowledge Of Formal Computation And Its Relationship To Languages
2. Classify The Relationships Between Language Classes And Regular Expression
3. Design Grammars And Recognizers For Different Formal Languages
4. Identify The Equivalence Of Languages Described By Pushdown Automata
5. Evaluate The Language Acceptability By Turing Machine
6. Understand The Basics Of Compiler

Unit I: Preliminaries And Finite State Machines

Introduction To Theory Of Computation- Automata, Alphabet, Symbol, String, Formal Languages, Deterministic Finite Automaton (DFA)- Definition, Representation, Acceptability Of A String And Language, Non-Deterministic Finite Automaton (NFA), Equivalence Of DFA And NFA, NFA With E-Transition, Equivalence Of NFA's With And Without E-Transition, Finite Automata With Output- Moore Machine, Mealy Machine, Equivalence Of Moore And Mealy Machine, Minimization Of Finite Automata. **06 Hours**

Unit II: Regular Languages

Definition And Examples. Conversion Of RE To FA, FA To RE, Algebraic Laws, Pumping Lemma For Regular Languages And Applications. Closure Properties Of Regular Languages Union, Concatenation, Complement, Intersection And Kleene Closure. Decidability- Decision Properties. **06 Hours**

Unit III: Context Free Grammar (CFG)

Definition, Derivations, Languages, Derivation Trees And Ambiguity, Regular Grammars-Right Linear And Left Linear Grammars, Conversion Of FA Into CFG And Regular Grammar Into FA, Simplification Of CFG, Normal Forms- Chomsky Normal Form (CNF), Greibach Normal Form (GNF),Chomsky Hierarchy, Programming Problems Based On The Properties Of Cfgs. **06 Hours**

Unit IV: Push Down Automata (PDA)

Introduction, Pushdown Automata (PDA), Transition Diagrams, Functions And Tables, Deterministic Push- Down Automata (DPDA) - Definition, Nondeterministic Pushdown Automata (NPDA), Equivalence Of Context Free Grammars And PDA, Properties Of Context Free Languages. Introduction To Post Machines (Pms). **06 Hours**

Unit V: Turing Machines

The Turing Machine Model And Definition Of TM, Language Acceptability Of Turing Machines, Techniques For Turing Machine Construction, Modifications Of Turing Machine, Composite And Iterative Turing Machines, Multi Tape Turing Machine, Multi Stack And Multi Track Turing Machine, Universal Turing Machine, Linear Bounded Automata, Church's Thesis, Halting Problem. **06 Hours**

Unit VI: Applications

Applications Of Regular Expressions, Lexical Analyzer, Text Editor, And Searching Using RE, Context Free Grammar, Basics Of Parsing Techniques, Application Of Leftmost And Rightmost Derivations During Parsing, Primitive Recursive Functions, Recursive And Recursively Enumerable Languages, Introduction To Natural Language Processing. **06 Hours**

Textbooks:

1. Vivek Kulkarni "Theory Computation" Oxford Higher Education
2. Theory Of Computer Science (Automata, Language & Computation) K. L. P. Mishra & N. Chandrasekaran, PHI Second Edition.
3. E.V. Krishnamurthy, "Theory Of Computer Science", EWP Publication.

Reference Books:

1. Hopcroft Ullman, "Introduction To Automata Theory, Languages &Computations, Narosa.
2. Daniel A. Cohen, "Introduction To Computer Theory", Wiley Publication.
3. Automata Theory, Languages, And Computation, John E. Hopcroft Cornell University, Rajeev Motwani Stanford University, Jeffrey D. Ullman Stanford University,3rd Edition.

List Of Assignments:

1. Study Of JFLAP Tool For Constructing FA.
2. Construct Regular Expressions Defined Over The Alphabet Σ= {A, B}, Which Denote The Given Languages.
3. Translate The Following Mealy Machine Into Its Equivalent Moore Machine.
4. Write A Context-Free Grammar (CFG) Which Generates The Language L Denoted By:(A+ B)*Bbb(A+ B)*.
5. Construct A PDA That Accepts The Language Defined By The Following Regular Grammar.
6. Design A TM To Recognize An Arbitrary String Divisible By 4, From Σ= {0, 1, 2}.

Project Based Learning:

1. Develop A Tool To Illustrate The Algorithm For Converting An Arbitrary NFA To A DFA .
2. Develop A Tool To Draw A Transition Diagram For Any Given DFA.
3. Approximation Algorithms
4. Greedy Algorithms.
5. Enumeration Of Finite Automata
6. Enumeration Of PDA
7. Enumeration Of Turing Machines
8. Ambiguous Grammars
9. Disambiguation Of Grammars
10. Enumeration Of Context-Free Languages
11. . Enumeration Of Turing Machines
12. . Universal Turing Machines.
13. Randomized Turing Machines
14. NP Complete Algorithm
15. Problem Solvability Using Reduction
16. Design Of TM To Emulate A Finite Automata
17. . Design Of TM To Emulate A PDA
18. Complexity Analysis Of Encryption Algorithms Using TM.
19. . Design Of TM To Perform Sorting

20. Design TM To Perform Searching.

Syllabus for Unit Test

Unit Test I	Unit No. I, II, III
Unit Test II	Unit No. IV, V, VI

System Programming and Operating System

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	03 Hours/Week	End Semester Examination:	60 Marks	Lecture:	03
Practical:	02 Hours/Week	Internal Assessment:	40 Marks	Practical:	01
		Term Work:	25 Marks		
		Oral:	25 Marks		
		Total:	150 Marks	Total	04

Course Objectives:

- To Help The Students Understand Functioning Of Various System Programs And Compiler, Loaders And Linkers.
- To Help Students For Different Concepts Of Operating System And Management With File System.

Prerequisite:

Knowledge Of Microprocessor Concepts And Assembly Language And Concept Of System Software, Application Software, Knowledge Of Input Output Devices And Its Usage.

Course Outcomes: On Completion Of The Course, Students Will Have The Ability To:

1. Understand Basic System Software And Its Functionality.
2. Identify And Analyze The Working Of Assemblers, Loaders And Linkers.
3. Compare And Evaluate Different Scheduling Algorithms.
4. Identify The Mechanism To Deal With Deadlock And Concurrency Issues.
5. Analyze Memory Management Policies And Virtual Memory.
6. Apply Different File Management Techniques

Unit I: Introduction To Systems Programming **06 Hours**

Need Of System Programming, Software Hierarchy, Types Of Software: System Software And Application Software, Machine Structure. Components Of System Software: Text Editors, Assembler, Macros, Compiler, Interpreter, Loader, Linker, Debugger, Device Drivers, Operating System.
Elements Of Assembly Language Programming, Structure Of An Assembler, Design Of Single And Two Pass Assemblers.

Unit II: Introduction To OS **06 Hours**

Architecture, Goals & Structures Of O.S, Basic Functions, Interaction Of O. S. & Hardware Architecture, System Calls, Batch, Multiprogramming. Multitasking, Time Sharing, Parallel, Distributed & Real -Time O.S.

Unit III: Process Management **06 Hours**

Process Concept, Process States, Process Control, Threads, Scheduling: Types Of Scheduling: Pre-Emptive, Nonredemptive, Scheduling Algorithms: FCFS, SJF, RR, Case Study On Unix /Linux OS.

Unit IV: Concurrency Control **06 Hours**

Concurrency: Interprocess Communication, Mutual Exclusion, Semaphores, Classical Problems Of Synchronization: Readers-Writers, Producer Consumer, And Dining Philosopher Problem. Deadlock: Principles Of Deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Case Study Related To Unit Contents.

Unit V: Memory Management **06 Hours**

Basics Of Memory Management, Swapping, Memory Allocation, Paging, Segmentation, Virtual Memory, Demand Paging, Page Replacement, Page Replacement Algorithms – Optimal FIFO, LRU, LRU Approximation, Allocation Of

Frames, Case Study Related To Unit Contents.

Unit VI: File Management **06 Hours**

Concepts, File Organization, File Directories, File Sharing, Record Blocking, Allocation Methods, Free Space Management, Comparative Study Of Traditional OS And Modern OS. Examples Of Modern Operating System.
Emerging Trends And Technologies Of OS

Textbooks:

1. System Programming By John J. Donovan, TATA Mcgraw-HILL Edition.

Reference Books:

1. Operating System Concepts, 9th Edition Peter B. Galvin, Greg Gagne, Abraham Silberschatz, John Wiley & Sons, Inc.
2. Operating Systems 5th Edition, William Stallings, Pearson Education India.
3. D. M. Dhamdere : "Systems Programming And Operating System", Tata Mcgraw Hill.

List Of Assignments:

1. Explain With Example Elements Of Assembly Language Programming.
2. Explain In Detail The Design Of Single And Two Pass Assembler.
3. What Is Operating System? Explain Types Of OS.
4. Explain Concept Of Process In Detail.
5. Explain Process Scheduling Algorithms. (FCFS, SJF, RR,)
6. Explain Interprocess Communication.
7. What Are The Different Classical Problems Of Synchronization.
8. Write A Note On Memory Management.
9. Explain Various Page Replacement Algorithms.
10. Give Examples Of Modern OS.

Project Based Learning:

1. Develop Heap Memory Manager In C
2. Design The Processes And Thread Management With Deadlock's, Synchronization
3. Design Preemptive Priority Scheduling Algorithm Implementation In Any Language.
4. Java Program To Analyze Page Fault For A Given Page Frame Using NRU With Paging.
5. The Project On Simulating The Multiprogramming Of A Specific Operating System And Dealing With CPU Scheduling And Job Scheduling.
6. Design The Project That Computes FCFS, SSTF, And SCAN Disk-Scheduling Algorithms
7. Operating Systems Mini-Project To Explore The Different Algorithms Of Main Memory Page Replacement
8. Develop Any One Project On One Or Two Pass Assemblers.
9. Design A Simple Language And Develop A Compiler For The Three-Address Code Generation And Evaluation Using Lex And Yacc.
10. Construct A Parser That Recognizes A Specific Language.

List Of Laboratory Exercises:

1. Design One Pass Assembler
2. Design Two Pass Assembler
3. Write A Program To Create Dynamic Link Library For Any Mathematical Operation And Write An Application Program To Test It
4. Write A Program Using Lex Specifications To Implement Lexical Analysis Phase Of Compiler To Count No. Of Words, Lines And Characters Of Given Input File.
5. Write A Program To Solve Classical Problem Of Synchronization Using Mutex And Semaphore.
6. Implement UNIX System Calls Like Ps, Fork, Join, Exec Family, And Wait For Process Management.
7. Implementation Of Various Scheduling Algorithm.

8. Implementation Of Banker's Algorithm.
9. Find Out The Page Fault Of Any Given String.
10. Implementation Of Various Page Replacement Algorithm.
11. Study Assignment On Process Scheduling Algorithms For Latest OS.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Computer Organization & Architecture

Teaching Scheme		Examination Scheme		Credit Scheme	
Hours/Week		Marks		Credits	
Lecture:	03	University Examination:	60	Lecture	03
		Internal Assessment:	40		
		Total	100 Marks	Total	03

Course Objective:

To Understand The Design Of Various Functional Units Of Computer System And To Explain Key Skills Of Constructing Cost-Effective Computer Systems.

Prerequisite:

Students Should Have Basic Knowledge Of Digital Electronics And Computer System.

Course Outcomes: On Completion Of The Course, Students Will Have The Ability To:

1. Identify Various Components Of A Computer And Their Interconnection.
2. Explain The Functions & Organization Of Various Blocks Of CPU.
3. Understand CPU Instruction Characteristics, Enhancement Features Of CPU.
4. Describe An Assortment Of Memory Types (With Their Characteristics) Used In Computer Systems And Basic Principle Of Interfacing Input And Output Devices.
5. Describe The I/O Organization And Interconnections.
6. Infer Parallel Processing And Multiprocessor Configuration.

Unit I: Structure Of Computer

Organization, Functions & Types Of Computer Units- CPU (Typical Organization, Functions, Types), Memory (Types & Their Uses In Computer), IO (Types & Functions) & System Bus (Address, Data & Control, Typical Control Lines, Multiple-Bus Hierarchies); Von Neumann & Harvard Architecture; Instruction Cycle **06 Hours**

Data Representation: Signed Number Representation, Fixed And Floating-Point Representations, Character Representation.

Unit II: Computer Arithmetic

Integer Addition And Subtraction, Ripple Carry Adder, Carry Look-Ahead Adder, Etc. Multiplication – Shift-And-Add, Booth Multiplier, Carry Save Multiplier, Etc. Division Restoring And Non- Restoring Techniques, Floating Point Arithmetic, IEEE 754 Format. **06 Hours**

Processor: Single Bus Organization Of CPU; ALU (ALU Signals, Functions & Types); Register (Types & Functions Of User Visible, Control & Status Registers Such As General Purpose, Address Registers, Data Registers, Flags, PC, MAR, MBR, IR) & Control Unit (Control Signals & Typical Organization Of Hard Wired & Microprogrammed CU). Micro Operations (Fetch, Indirect, Execute, Interrupt) And Control Signals For These Micro-Operations.

Unit III: Instruction Set Architecture Of A CPU

Elements Of Machine Instruction; Instruction Representation (Opcode& Mnemonics, Assembly Language Elements), Instruction Format & 0-1-2-3 Address Formats, Types Of Operands, Addressing Modes; Instruction Types Based On Operations (Functions & Examples Of Each), RTL Interpretation Of Instructions, Exceptions; Instruction Pipelining(Operation & Speed Up) Complex Instruction Set Computer (CISC) , Reduced Instruction Set Computer (RISC), Key Characteristics Of RISC& CISC, CISC Vs RISC. **06 Hours**

Interrupt: Its Purpose, Types, Classes & Interrupt Handling (ISR, Multiple Interrupts)

Unit IV: Memory System Design

Semiconductor Memory Technologies. **06 Hours**

Memory Organization: Memory Interleaving, Concept Of Hierarchical Memory Organization, Cache Memory, Cache Size Vs. Block Size, Mapping Functions, Replacement Algorithms, Write Policies.

Unit V: INPUT OUTPUT

I/O Interface, Programmed IO, Memory Mapped IO, Interrupt Driven IO, DMA.**06 Hours**
Multiprocessors: Characteristics Of Multiprocessors, Interconnection Structures, Inter Processor Arbitration, Inter Processor Communication And Synchronization, Cache Coherence.

Programs And Processes – Role Of Interrupts In Process State Transitions, I/O Device Interfaces – SCSI, USB

Unit VI: Pipelining

Basic Concepts Of Pipelining, Throughput And Speedup, Pipeline Hazards. **06 Hours**

Parallel Processors: Introduction To Parallel Processors, Concurrent Access To Memory And Cache Coherency

Textbooks:

1. Computer System Architecture M. M. Mano: 3rd Ed., Prentice Hall Of India, New Delhi, 1993.
2. Computer Organization And Design: The Hardware/Software Interface, David A. Patterson And John L. Hennessy.
3. Computer Organization And Embedded Systems, Carl Hamacher.

Reference Books:

1. Computer Architecture And Organization, John P. Hayes.
2. Computer Organization And Architecture: Designing For Performance, William Stallings

List Of Assignments:

1. Booth's Algorithm For Multiplication
2. Restoring And Non-Restoring Division
3. Fixed Point And Floating-Point Representation
4. Programmer's Model Of 80386
5. Hardwired And Micro-Programmed Design Approaches.
6. Characteristics Of Memory System
7. Cache Organization And Address Mapping
8. Virtual Memory And Replacement Algorithms
9. Calculating Throughput And Speed In Pipelining
10. Multiprocessor Architecture

Project Based Learning:

1. Automatic Night Lamp With Morning Alarm
2. Traffic Light With Sensor + 7segment
3. Multi Pattern Running Lights.
4. Washing Machine
5. Simple Lock Using Keypad And 7 Segment.
6. Electronic Quiz Table
7. Electronic Digital Clock
8. Temperature Controller

9. Plant Irrigation System
10. Car Parking Management
11. Customer Counter For Supermarket
12. Electronic Queue Management System In Food Stall
13. Safety Box
14. Shop Lot Automatic Door With 7segment Display
15. Bank Queue Management System
16. Water Level Controller
17. Automatic Home System
18. Commuter System
19. Automatic Room Light Control
20. Elevator Control System

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Computer Graphics and Multimedia

Teaching Scheme		Examination Scheme		Credit Scheme	
					Credits
Lecture:	**03 Hours/Week**	**End Semester Examination:**	**60 Marks**	**Lecture :**	**03**
Practical:	**02 Hours/Week**	**Internal Assessment:**	**40 Marks**	**Practical:**	**01**
		Term Work:	**25 Marks**		
		Total	**125 Marks**	**Total:**	**04**

Course Objectives:

1. To Introduce The Principles Of Computer Graphics And The Components Of A Graphics System.
2. To Introduce Basic Algorithms For Drawing Line, Circle And Curves.
3. To Develop Understanding Of The Basic Principles Of 2D And 3D Computer Graphics And How To Transform The Shapes To Fit Them As Per The Picture Definition.
4. To Introduce Multimedia Architecture And Hardware.
5. To Introduce Multimedia File Formats.

Prerequisite:

Knowledge Of C Programming Language, Linear Algebra.

Course Outcomes: On Completion Of The Course, Students Will Have The Ability To:

1. Identify The Basic Terminology Of Computer Graphics And Apply Scan Conversion Of Points, Lines, Circle For Problem Solving.
2. Create Effective Programs Using Concepts Of Polygon, Fill Polygon, Curves, Fractals.
3. Analyze Techniques Of Geometrical Transforms Objects In 2 Dimensional.
4. Evaluate The Core Concepts Of Computer Graphics, Including Transformation In Three Dimensions, And Projection.
5. Understand The Concepts Of Multimedia
6. Analyze The Concepts Of Hypermedia

Unit I: Introduction To Computer Graphics And Devices

Introduction To Computer Graphics, Applications Of Computer Graphics **06 Hours**

Graphic Devices : CRT, Raster Scan & Random Scan Displays, Color CRT, Beam Penetration CRT, The Shadow -Mask CRT, Direct View Storage Tube, Tablets, Display Processor, Display File Structure, Graphic Mode Commands

Graphics Output Primitives:

Point And Lines,

Line Drawing Algorithms: DDA Line Drawing Algorithm, Bresenham Line Drawing Algorithm,

Circle Drawing: DDA, Midpoint, Bresenham Circle Drawing Algorithm,

Unit II: Polygons

Polygons And Its Types, Inside Test, **06 Hours**

Polygon Filling Methods: Seed Fill – Flood Fill And Boundary Fill, Scan-Line Fill Algorithms; Character Generation, Antialiasing**.**

Curves And Fractals

Curves: Introduction, Interpolation And Approximation, Spline Interpolation Methods , Bezier Curves, B-Splines.

Fractals: Introduction, Classification, Fractal Dimension, Fractal Dimension And Surfaces, Hilbert Curve, Koch Curve.

Unit III: Two-Dimensional Graphics 2D Transformations

Translation, Scaling, Rotation, Reflection And Shearing, Matrix **06 Hours**

Representation And Homogeneous Coordinate System, Composite Transformations.

Windowing: Concept Of Window And Viewport, Viewing Transformations

Line Clipping: Cohen Sutherland Method Of Line Clipping

Polygon Clipping: Sutherland Hodgeman Method For Convex And Concave Polygon Clipping.

Unit IV: Three-Dimensional Graphic 3D Transformation

Translation, Scaling, Rotation About X, Y, Z & Arbitrary Axis, And Reflection About XY, YZ, XZ & Arbitrary Plane. **06 Hours**

Projections: Types Of Projections- Parallel, Perspective

Parallel: Oblique – Cavalier, Cabinet, Orthographic – Isometric, Diametric, Trimetric

Perspective: Vanishing Points As 1 Point, 2 Point And 3 Point

Unit V: Multimedia System Design & Multimedia File Handling

Multimedia Basics, Multimedia Applications, Multimedia System Architecture, Evolving Technologies For
Multimedia, Defining Objects For Multimedia Systems, Multimedia Data Interface Standards. Compression And Decompression, Data And File Format Standards, Multimedia I/O Technologies, Digital Voice And Audio, Video Image And Animation, Full Motion Video, Storage And Retrieval Technologies.

Unit VI: Hypermedia

Multimedia Authoring And User Interface-Hypermedia Messaging -
Mobile Messaging – Hypermedia Message Component – Creating Hypermedia Message – Integrated Multimedia Message Standards– Integrated Document Management – Distributed Multimedia Systems.

Textbooks:

1. Donald Hearn And Pauline Baker M, Computer Graphics", Prentice Hall, New Delhi, Second Edition.
2. Andleigh, P. K And Kiranthakrar, Multimedia Systems And Design, PHI, 2015.

Reference Books:

1. Foley, Vandam, Feiner And Hughes, Computer Graphics: Principles And Practice, 2nd Edition, Pearson Education, 2003.
2. Peter Shirley, Michael Ashikhmin, Michael Gleicher, Stephen R Marschner, Erik Reinhard, Kelvinsung, And AK Peters, Fundamentals Of Computer Graphics, CRC Press, 2010.

List Of Assignments:

1. Study On Any Latest Data Generating Device In Computer Graphics .
2. Describe In Detail Any One Color Model Process.
3. Using The Determinant To Calculate The Signed Areas, Determine Whether The Point $\boldsymbol{P}$=[23]T Is In The Triangle Formed By The Points $\boldsymbol{S}$=[22]T, $\boldsymbol{T}$=[51]T And $\boldsymbol{R}$=[35]T. Show All Of Your Work.
4. Elaborate In Detail The Any One Curve Generation Methods.
5. Describe How Multimedia System Architecture Process Is Used In Real Time.

Project Based Learning:

1. Helicopter Game
2. Sinking Ship
3. Scientific Calculator
4. Traditional Wall Clock

5. Tower Of Hanoi Game
6. Windmill
7. Steam Engine
8. Traffic Signal
9. Aquarium
10. Prepare A Powerpoint Presentation
11. Mobile App For Online Shopping
12. Arrival And Departure Of The Train With Announcement And Signal
13. Mobile Application For Online Tour Guidance App
14. Create A Small Video On The Given Topic
15. Story Tell Mobile App

List Of Laboratory Exercises:

1. Study Of Fundamental Graphics Functions.
2. Implementation Of Line Drawing Algorithms: DDA Algorithm, Bresenham's Algorithm.
3. Implementation Of Circle Drawing Algorithms: Bresenham's Algorithm, Mid-Point Algorithm. Programs On 2D And 3D Transformations.
4. Write A Program To Implement Cohen Sutherland Line Clipping Algorithm.
5. Write A Program To Implement Sutherland Hodgeman Polygon Clipping Algorithm.
6. Write A Program To Implement DDA Curve
7. Write A Program To Implement Bezier Curve
8. Write A Program To Implement Fractal

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

COMPUTER NETWORKS

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	03 Hours/Week	End Semester Examination:	60 Marks	Lecture	03
Practical:	02 Hours/Week	Internal Assessment:	40 Marks	Practical	01
		Term Work	25 Marks		
		Practical	25 Marks		
		Total	150 Marks	Total	04

Course Objective:

1. To Understand The Fundamental Concepts Of Networking And Technologies
2. To Learn Different Techniques For Framing, Error Control, Flow Control And Routing
3. To Learn Different Layer Protocols In The Network Layer
4. To Understand Modern Network Architectures With Respect To Design And Performance.

Prerequisite:

Computer Systems Workshop Technology

Course Outcomes: On Completion Of The Course, Students Will Have The Ability To:

1. Understand Fundamental Concepts Of Computer Networks, Architectures, Protocols And Technologies
2. Remember The Working And Functions Of Data Link Layer
3. Analyze The Working Of Different Routing Protocols And Mechanisms
4. Implement Client-Server Applications Using Sockets
5. Illustrate Role Of Application Layer With Its Protocols, Client-Server Architectures
6. Apply Concepts Of Network Security.

Unit I : Basics Of Computer Networks **06 Hours**

Introduction To Computer Network, Types Of Computer Network,Ad-Hoc Network, Network Architectures: Client-Server; Peer To Peer; Distributed And SDN, OSI Model, TCP/IP Model, Topologies: Star And Hierarchical; Design Issues For Layers, Transmission Mediums: Cat5, 5e, 6, OFC And Radio Spectrum, Network Devices: Bridge, Switch, Router, Brouter And Access Point.

Unit II : Data Link Layer **06 Hours**

Design Issues, Error Detection And Correction, Elementary Data Link Protocols, Sliding Window Protocols, Example Data Link Protocols - HDLC, The Data Link Layer In The Internet. The Medium Access Sublayer: Channel Allocations Problem, Multiple Access Protocols, Ethernet, Data Link Layer Switching, Wireless LAN, Broadband Wireless, Bluetooth.

Unit III : **Network Layer** **06 Hours**

Switching Techniques, IP Protocol, Ipv4 And Ipv6 Addressing Schemes, Subnetting, Address Mapping – ARP, RARP, BOOTP And DHCP–Delivery ,Routing Protocols: Distance Vector, Link State, Path Vector, Routing In Internet: RIP ,OSPF, BGP.

Unit IV : Transport Layer **06 Hours**

Transport Service, Transport Layer Protocols For Flow Control, Elements Of Transport Protocols, Multiplexing, Congestion Control, Example Protocols: Udp, Tcp.

Unit V : Application Layer **06 Hours**

Domain Name System (Dns), Hyper Text Transfer Protocol (Http), Email: Smtp,

Mime, Pop3, Webmail, Ftp, Telnet, Dynamic Host Control Protocol (Dhcp), Www, Http, Snmp.

Unit VI : Network Security **06 Hours**

Network Security: Electronic Mail, Directory Services And Network Management, Basic Concepts Of Cryptography.

Textbooks:

1. Fourauzan B.,"Data Communications And Networking", 5 Th Edition, Tata Mcgraw-Hill, Publications, ISBN:0–07 – 058408 – 7
2. Andrew S. Tanenbaum, Computer Networks, 5th Edition, Pearson India, 2012.

Reference Books:

1. Kurose, Ross, “Computer Networking A Top Down Approach Featuring The Internet”, Pearson, ISBN-10: 0132856204
2. L. Peterson And B. Davie, “Computer Networks: A Systems Approach”, 5th Edition, Morgan-Kaufmann, 2012.
3. Douglas E. Comer & M.S Narayanan,”Computer Network & Internet”, Pearson Education

List Of Assignments:

1. Study Of LAN Environment.
2. Explain OSI Model In Detail.
3. Explain Error Detection And Correction Methods With Example.
4. Study The Concept Of Subnetting At Network Layer.
5. Discuss Switching Techniques In Detail.
6. Explain Multiplexing Methods In Detail.
7. Assignment On Application Layer
8. Assignment On Bandwidth Utilization Techniques.

List Of Laboratory Exercises:

1. Setup A Wired LAN Using Switch, Router And Then IP Switch Of Minimum Four Computers, Configuration Machine Using IP Addresses, Testing Using PING Utility Using Network Simulation Tool Cisco Packet Tracer.
2. Write A Program For Error Detection And Correction For 7/8 Bits ASCII Codes Using Hamming Codes And CRC.
3. Write A Program To Simulate Go Back N And Selective Repeat Modes Of Sliding Window Protocol In Peer To Peer Mode.
4. Write A Program To Demonstrate Subnetting And Find The Subnet Masks
5. Configure RIP/OSPF/BGP Using Packet Tracer
6. Write A Program For DNS Lookup. Given An IP Address Input, It Should Return URL And Vice-Versa
7. Write A Program Using TCP Socket For Wired Network.
8. Write A Program Using UDP Sockets To Enable File Transfer (Script, Text, Audio And Video One File Each) Between Two Machines
9. Use Network Simulator NS2 To Implement: Monitoring Traffic For The Given Topology
10. Study Of Installation And Configuration Of DHCP Server

Project Based Learning:

1. Chat Application
2. Multiuser Chat Application
3. Network Monitoring System
4. Intruder Detection System
5. Computing Shortest Path Between Nodes
6. Client-Server Based Instant Messenger
7. File Transfer Protocol

8. Network Security Protocol With Cryptography
9. Peer To Peer Resource Monitoring System
10. Distance-Vector-Routing-And-Flow-Control-Simulator
11. Link State Routing Protocol To Find Suitable Path For Transmission.
12. Analyse Following Packet Formats Captured Through Wireshark For Wired Network

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Skill Lab IV (Python Programming)

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	-	End Semester Exam:	-	Lecture	00
Practical:	02 Hours/Week	Internal Assessment:	-	Practical	01
		Term Work	25 Marks		
		Practical	25 Marks		
		Total	50 Marks	Total	01

Course Objective:

1. To Teach Students Basic Of Python Programming.
2. To Learn Control Statements And Data Structures In Python.
3. To Understand And Demonstrate Use Of Functions And Classes In Python.
4. To Learn File Handling And Different Packages In Python For Real Time Datasets.
5. To Learn Implementation Of Classes, Inheritance, Polymorphism In Python.
6. To Learn Networking And GUI In Python.

Prerequisite:

Basic Use Of Computer And Fundamentals Of Programming Language.

Course Outcomes: On Completion Of The Course, Students Will :

1. Understand Basics Of Python Language.
2. Apply Functions Using Looping Structures.
3. Create And Manipulate File Operations In Python. Apply Different Packages In Python On Real Time Datasets.
4. Understand Concepts Of Classes And Objects.
5. Implementation Of Inheritance And Polymorphism.
6. Design GUI And Create Networking Interface Between Client And Server.

Unit I: Introduction To Python Programming **04 Hours**

Python Interpreter, Variables, Operators (Arithmetic, Bitwise, Logical, Assignment), Data Types (List, Tuple, Dictionary And Set) And Their Important Methods, String Operations And Slicing.

Unit II: Control Flow Statements And Functions **04 Hours**

Control Flow Statements: Control Flow Statements, Looping Structures Using Pass, Continue, Break Statements.
Functions: Defining Functions And Arguments, Passing Parameters To A Function, Lambda Function.

Unit-III: File Handling And Data Science Packages **04 Hours**

File Handling: Concept Of Files, File Opening In Various Modes And Closing Of A File, Different Access Modes, Reading From A File, Writing Onto A File, Important File Handling Functions E.G Open(), Close(), Read(), Readline() Etc.
Data Science Packages: Numpy, Pandas And Matplotlib Etc.

Unit-IV: Oops Concepts **04 Hours**

OOPS In Python: Features Of Object-Oriented Programming System (OOPS), Classes And Objects, Encapsulation, Abstraction, Inheritance, Polymorphism, Constructors And Destructors .
Classes And Objects: Creating A Class, The Self-Variable, Types Of

Variables, Namespaces, Types Of Methods, Instance Methods, Class Methods, Static Methods, Passing Members Of One Class To Another Class, Inner Classes.

Unit-V: Inheritance And Polymorphism **04 Hours**

Inheritance In Python, Types Of Inheritance- Single Inheritance, Multilevel Inheritance, Hierarchical Inheritance, Multiple Inheritance, Constructors In Inheritance, Overriding Super Class Constructors, Super() Method, Method Resolution Order (Mro), Polymorphism, Duck Typing, Operator Overloading, Method Overloading, Method Overriding. Abstract Classes And Interfaces: Abstract Class, Abstract Method, Interfaces In Python, Abstract Classes Vs. Interface.

Unit VI: Networking And GUI **04 Hours**

Networking:Protocols,Server:
Client Architecture, TCP/IP And UDP Communication.
Graphical User Interface: Creating A GUI In Python, Widget Classes, Working With Fonts And Colours, Working With Frames, Layout Manager, Event Handling.
Textbooks:

1. Starting Out With Python Plus Myprogramminglab With Pearson Etext 3rd Edition , Tony Gaddis
2. Introduction To Python Programming, Chapman And Hall/CRC 1 Edition By Gowrishankar, Veena
3. Fundamentals Of Python Programming By Richard L. Halterman

Reference Books:

1. Python Crash Course By Eric Matthes
2. Learning Python By Mark Lutz

List Of Experiments:

1. Develop A Program To Read The Student Details Like Name, PRN, Marks In Three Subjects. Display The Student Details, Total Marks And Percentage With Suitable Messages.
2. Develop A Program To Read The Name And Year Of Birth Of A Person. Display If The Person Is A Senior Citizen.
3. Develop A Program To Generate Fibonacci Sequence Of Length N. Take Use Input For N.
4. Write A Program To Print Factorial Of A Number.
5. Accept A Multi Digit Number From User And Find Put The Frequency Of Each Digit With Suitable Message.
6. Create Data Structures Such As List, Tuple, Dictionary And Set In Python And Perform Important Functions On These Data Structures.
7. Implement A Python Program To Show File Operations.
8. Demonstrate Use Of Various Python Packages Such As Numpy, Pandas And Matplotlib.
9. Wrte A Program To Python Program To Implement Concepts Of OOP Such As:
   A) Abstract Methods And Classes
   B) Interfaces
   C) Types Of Methods
10. Write A Program To Python Program To Implement Concepts Of OOP Such As
    A) Inheritance
    B) Polymorphism
11. Write A Python Program To Create Server-Client And Exchange Basic Information
12. Write A GUI Program In Python To Design Application That Demonstrates
    A) Different Fonts And Colors
    B) Different Layout Managers
    C) Event Handling