CURRICULUM

for

UNDERGRADUATE DEGREE COURSES IN

Computer Engineering (Third Year)

(Engineering & Technology)

[Proposed from 2020-21]

for

Swami Ramanand Thirth Marathwada University Nanded

SEMESTER – V (Third Year) **Branch/Course – Computer Engineering**

SI. No.	Category	Code	Course Title	Hours per Week				Marking Scheme				Theory Total
				L	T	P	CR	PR/OR (#)	PR/OR (@)	MSE	ESE	
1.	Engineering Science Courses	ESC 501	Signals & Systems	3	1	0	4	0	0	30	70	100
2.	Professional Core Courses	PCC-CS 502	Database Management Systems	3	1	2	5	25	25	30	70	150
3.	Professional Core Courses	PCC-CS 503	Formal Language & Automata Theory	3	1	2	5	25	25	30	70	150
4.	Professional Core Courses	PCC-CS 504	Object Oriented Programming	3	0	2	4	25	25	30	70	150
5.	Professional Elective Courses	PEC 505	Elective-I	3	0	2	4	25	25	30	70	150
6.	Mandatory Courses(Non-Credit)	MC 506 **($)**	Constitution of India/ Essence of Indian Traditional Knowledge	2	0	0	0	0	50	0	0	50
7.	Humanities and Social Sciences including Management Courses	HSMC 507	Seminar II	0	0	2	1	0	50	0	0	50
8.	Professional Core courses	PCC-CS 508	NPTEL COURSE I	0	0	2	1	0	0	0	0	00
Total				**17**	**3**	**12**	**24**	**100**	**200**	**150**	**350**	**800**
Grand Total				**32**								

Symbols to remember: - @ - Internal Assessment, # - External Assessment, $ - Audit Pass

T – Theory , P– Practical, T – Tutorial , CR – Credit , PR/OR – Practical/Oral, MSE – Minor Semester Examination, ESE – End Semester Examination.

Dr. U.V. Kulkarni
Shri Guru Gobind Singhji Institute of Engineering and Technology Vishnupuri, Nanded.
(Chairman)

Dr. G.V. Chaudhari
Director
School of Computational Sciences
S.R.T.M.U, Nanded.

Dr. J.V. Megha
Shri Guru Gobind Singhji Institute of Engineering and Technology Vishnupuri, Nanded.

Mr. Wahi G.S.
Gramin College of Engineering Vishnupuri, Nanded.

Mr. Maske P.S.
Gramin College of Engineering Vishnupuri, Nanded.

Dean Engineering
SRTM University Nanded

List of Professional and Open Electives:-

Professional Elective I:-

1. Web and Internet
2. Graph Theory
3. Software Engineering

Dr. U.V. Kulkarni
Shri Guru Gobind Singhji
Institute of Engineering
and Technology
Vishnupuri, Nanded.
(Chairman)

Dr. G.V. Chaudhari
Director
School of Computational
Sciences
S.R.T.M.U, Nanded.

Dr. J.V. Megha
Shri Guru Gobind Singhji
Institute of Engineering
and Technology
Vishnupuri, Nanded.

Mr. Wahi G.S.
Gramin College
of Engineering
Vishnupuri, Nanded.

Mr. Maske P.S.
Gramin College
of Engineering
Vishnupuri, Nanded.

Dean Engineering
SRTM University Nanded

SEMESTER – VI (Third Year) **Branch/Course – Computer Engineering**

SI. No.	Category	Code	Course Title	Hours per Week				Marking Scheme				Theory Total
				L	T	P	CR	PR/OR (#)	PR/OR (@)	MSE	ESE	
1.	Professional Core Courses	PCC-CS 601	Compiler Design	3	0	2	4	25	25	30	70	150
2.	Professional Core Courses	PCC-CS 602	Computer Networks	3	0	2	4	25	25	30	70	150
3.	Professional Elective Courses	PEC 603	Elective-II	3	0	2	4	25	25	30	70	150
4.	Professional Elective Courses	PEC 604	Elective-III	3	0	2	4	25	25	30	70	150
5.	Open Elective Courses	OEC 605	Open Elective –I	3	0	0	3	0	0	15	35	50
6.	Project	PROJ-CS 606	Project-I (Mini Project)	0	0	4	2	50	50	0	0	100
7.	Humanities and Social Sciences including Management Courses	HSMC 607	Seminar-III	0	0	2	1	0	25	0	0	25
8.	Humanities and Social Sciences including Management Courses	HSMC 608	Technical & Competitive Exams Skill	0	0	2	1	0	25	0	0	25
9.	PROFESSIONAL CORE COURSE	PCC-CS 609	NPTEL Course II	0	0	2	1	0	00	0	0	00
Total				**15**	**0**	**18**	**24**	**150**	**200**	**135**	**315**	**800**
Grand Total				**33**								

Symbols to remember: - @ - Internal Assessment, # - External Assessment, $ - Audit Pass

T – Theory , P– Practical, T – Tutorial , CR – Credit , PR/OR – Practical/Oral, MSE – Minor Semester Examination, ESE – End Semester Examination. *** *(Note: - Internship after 6th semester examination for 6 weeks)* *******

Dr. U.V. Kulkarni
Shri Guru Gobind Singhji
Institute of Engineering
and Technology
Vishnupuri, Nanded.
(Chairman)

Dr. G.V. Chaudhari
Director
School of Computational
Sciences
S.R.T.M.U, Nanded.

Dr. J.V. Megha
Shri Guru Gobind Singhji
Institute of Engineering
and Technology
Vishnupuri, Nanded.

Mr. Wahi G.S.
Gramin College
of Engineering
Vishnupuri, Nanded.

Mr. Maske P.S.
Gramin College
of Engineering
Vishnupuri, Nanded.

Dean Engineering
SRTM University Nanded

List of Professional and Open Electives:-

Professional Elective II:-

1. Advanced Computer Architecture
2. Advanced Operating Systems

Professional Elective III:-

1. Artificial Intelligence
2. Data Mining

Open Elective –I :

1. Soft Skills and Interpersonal Communication
2. ICT for development

Dr. U.V. Kulkarni
Shri Guru Gobind Singhji
Institute of Engineering
and Technology
Vishnupuri, Nanded.
(Chairman)

Dr. G.V. Chaudhari
Director
School of Computational
Sciences
S.R.T.M.U, Nanded.

Dr. J.V. Megha
Shri Guru Gobind Singhji
Institute of Engineering
and Technology
Vishnupuri, Nanded.

Mr. Wahi G.S.
Gramin College
of Engineering
Vishnupuri, Nanded.

Mr. Maske P.S.
Gramin College
of Engineering
Vishnupuri, Nanded.

Dean Engineering
SRTM University Nanded

ESC 501	Signals & Systems	3L:1T:0P	4 Credits

Course Contents:-

Module 1: Introduction to Signals and Systems: (08 Hrs)

Signals and systems as seen in everyday life, and in various branches of engineering and science. Signal properties: periodicity, absolute integrability, determinism and stochastic character. Some special signals of importance: the unit step, the unit impulse, the sinusoid, the complex exponential, some special time-limited signals; continuous and discrete time signals, continuous and discrete amplitude signals. System properties: linearity: additivity and homogeneity, shift-invariance, causality, stability, realizability. Examples.

Module 2: Behaviour of continuous and discrete-time LTI systems: (08 Hrs)

Impulse response and step response, convolution, input-output behaviour with aperiodic convergent inputs, cascade interconnections. Characterization of causality and stability of LTI systems. System representation through differential equations and difference equations. State-space Representation of systems. State-Space Analysis, Multi-input, multi-output representation. State Transition Matrix and its Role. Periodic inputs to an LTI system, the notion of a frequency response and its relation to the impulse response.

Module 3: Fourier Transform: (06 Hrs)

Fourier series representation of periodic signals, Waveform Symmetries, Calculation of Fourier Coefficients. Fourier Transform, convolution/multiplication and their effect in the frequency domain, magnitude and phase response, Fourier domain duality. The Discrete Time Fourier Transform (DTFT) and the Discrete Fourier Transform (DFT). Parseval's Theorem.

Module 4: Laplace and z- Transforms: (06 Hrs)

Review of the Laplace Transform for continuous time signals and systems, system functions, poles and zeros of system functions and signals, Laplace domain analysis, solution to differential equations and system behaviour. The z-Transform for discrete time signals and systems, system functions, poles and zeros of systems and sequences, z-domain analysis.

Module 5: Sampling and Reconstruction: (06 Hrs)

The Sampling Theorem and its implications. Spectra of sampled signals. Reconstruction: ideal interpolator, zero-order hold, first-order hold. Aliasing and its effects. Relation between continuous and discrete time systems.

Module 6: Applications of signal and system: (06 Hrs)

theory: modulation for communication, filtering, feedback control systems.

Course Outcomes:

At the end of this course, students will demonstrate the ability to:

1. Understand the concepts of continuous time and discrete time systems.
2. Analyze systems in complex frequency domain.
3. Understand sampling theorem and its implications.

References Books:

1. A. V. Oppenheim, A. S. Willsky and S. H. Nawab, “Signals and systems”, Prentice Hall India, 1997.
2. J. G. Proakis and D. G. Manolakis, “Digital Signal Processing: Principles, Algorithms, and Applications”, Pearson, 2006.
3. H. P. Hsu, “Signals and systems”, Schaum’s series, McGraw Hill Education, 2010.
4. S. Haykin and B. V. Veen, “Signals and Systems”, John Wiley and Sons, 2007.
5. A. V. Oppenheim and R. W. Schafer, “Discrete-Time Signal Processing”, Prentice Hall, 2009.
6. M. J. Robert “Fundamentals of Signals and Systems”, McGraw Hill Education, 2007.
7. B. P. Lathi, “Linear Systems and Signals”, Oxford University Press, 2009.

Page **7** of **50**

PCC-CS 502	Database Management System	3L:1T:2P	5 Credits

Course objectives:

1) Identify the basic concepts and various data model used in database design.
2) Formulate query, using SQL, solutions to a broad range of query and data update problems.
3) Recognize/ identify the purpose of query processing and optimization.
4) Demonstrate basic query evaluation.

Module 1 Introduction: **(02 Hrs)**

Definition Language (DDL), and Data Manipulation Language (DML).

Module 2: Data models: **(04 Hrs)**

Entity-relationship model, network model, relational and object oriented data models, integrity constraints, data manipulation operations.

Module 3: Relational query languages: **(06 Hrs)**

Relational algebra, Tuple and domain relational calculus, SQL3, DDL and DML constructs, Open source and Commercial DBMS - MYSQL, ORACLE, DB2, SQL server.

Module 4: Relational database design: **(05 Hrs)**

Domain and data dependency, Armstrong's axioms, Normal forms, Dependency preservation, Lossless design.

Module 5: Query processing and optimization: **(06 Hrs)**

Evaluation of relational algebra expressions, Query equivalence, Join strategies, Query optimization algorithms.

Module 6: Storage strategies: **(04 Hrs)**

Indices, B-trees, hashing.

Module 7: Transaction processing: **(06 Hrs)**

Concurrency control, ACID property, Serializability of scheduling, Locking and timestamp based schedulers, Multi-version and optimistic Concurrency Control schemes, Database recovery.

Module 8: Database Security: **(04 Hrs)**

Authentication, Authorization and access control, DAC, MAC and RBAC models, Intrusion detection, SQL injection.

Module 9: Advanced topics: **(04 Hrs)**

Object oriented and object relational databases, Logical databases, Web databases, Distributed databases, Data warehousing and data mining.

List of Practical's:-

1. For a given query write relational algebra expressions for that query and optimize the developed expressions.

2. For a given specification of the requirement design the databases using E R method and normalization.
3. For a given specification construct the SQL queries for Open source and Commercial DBMS -MYSQL, ORACLE, and DB2.
4. For a given query optimize its execution using Query optimization algorithms.
5. Implement the isolation property, including locking, time stamping based on concurrency control and Serializability of scheduling.
6. Data Definition Language Commands.
7. Data Manipulation Language Commands.
8. Data Control Language, Transfer Control Language Commands.
9. In Built Functions.
10. Nested Queries and Join Queries.
11. Set operators.
12. Views.
13. Control Structure.
14. Procedure and Function.
15. Trigger.
16. Front End Tools.
17. Menu Design.
18. Report Generation Database Design and Implementation Payroll Processing.

Course Outcomes:

1. To understand the different issues involved in the design and implementation of a database system.
2. To study the physical and logical database designs, database modelling, relational, hierarchical, and network models.
3. To understand and use data manipulation language to query, update, and manage a database.
4. To develop an understanding of essential DBMS concepts such as: database security, integrity, concurrency, distributed database, and intelligent database, Client/Server (Database Server), Data Warehousing.
5. To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modelling, designing, and implementing a DBMS.

References Books:

1 "Database System Concepts", 6th Edition by Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill.

2 "Principles of Database and Knowledge – Base Systems", Vol 1 by J. D. Ullman, Computer Science Press.

3 "Fundamentals of Database Systems", 5th Edition by R. Elmasri and S. Navathe, Pearson Education.

4 "Foundations of Databases", Reprint by Serge Abiteboul, Richard Hull, Victor Vianu, Addison-Wesley.

PCC-CS 503	Formal Language & Automata Theory	3L:1T:2P	5 Credits

Objectives of the course

1. Develop a formal notation for strings, languages and machines.
2. Design finite automata to accept a set of strings of a language.
3. Prove that a given language is regular and apply the closure properties of languages.
4. Design context free grammars to generate strings from a context free language and convert them into normal forms.
5. Prove equivalence of languages accepted by Push Down Automata and languages generated by context free grammars.
6. Identify the hierarchy of formal languages, grammars and machines.
7. Distinguish between computability and non-computability and Decidability and undesirability.

Course contents

Module 1 Introduction to Set Theory: (06 Hrs)

Set - Definition, Finite and infinite set, Countability of a set, Cardinality of a set, set operations, Closure of a set. Basic concepts - Symbols, Alphabet, String, Language, Difference between natural and formal language. Recursive definition of regular expression, Algebraic laws for regular expressions, Equivalence of regular expressions with finite automata, Regular set.

Module 2 Finite Automata (FA): (08 Hrs)

Deterministic and Non-deterministic FA, Equivalence of NFAs and DFAs, NFA with ε-moves, Minimization of DFA, Equivalence of regular expression and FA, FA with output - Definition, Models, Interconversion, Application of FA – Text search, Recognizing a Set of Keywords.

Module 3 Grammars & Production systems : (08 Hrs)

Definition of CFG, Designing context-free grammars, Derivation trees, Ambiguous grammar,inherently ambiguous grammar, removing ambiguity, Chomsky hierarchy, Regular grammar - definition, Left linear & right linear Regular Grammar, Inter-conversion between left linear and right linear regular grammar. Regular grammar and Finite Automata,, Simplification of CFG, Application of CFG.

Module 4 Properties of Context Free Languages: **(06 Hrs)**

Context Free Language (CFL) - definition, Normal forms - Chomsky Normal Form (CNF), Griebach Normal Form (GNF), closure properties of CFL, Pumping lemma for CFL, CYK algorithm for testing membership in a CFL.

Module 5 Pushdown Automata: (05 Hrs)

Definition - PDA, Non-deterministic PDA, Acceptance by final state and empty store, Construction of PDA, Equivalence between pushdown automata and context-free grammars, Multi-stack PDA, Introduction to Grammar systems and distributed Automata – CD (Cooperative distributed) grammar system, PC (Parallel Computing) Grammars.

Module 6 Turing Machines: (05 Hrs)

TM construction, Composite and Iterative TM, Universal TM, TM as enumerator, Variations of TM – Two-way Infinite Tape TM, Multi-tape TM

Module 7 Linear Bounded Automata: (04 Hrs)

Recursive sets, Recursively Enumerable Sets, Church's Turing Hypothesis, Halting problem. Undeciadability- Codes for Turing Machines, Notion of Undecidable problems, Rice's theorem, Post's correspondence problem (PCP).

Suggested books

1. John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman, Introduction to Automata Theory, Languages, and Computation, Pearson Education Asia.

Suggested reference books:

1. Harry R. Lewis and Christos H. Papadimitriou, Elements of the Theory of Computation, Pearson Education Asia.
2. Dexter C. Kozen, Automata and Computability, Undergraduate Texts in Computer Science, Springer.
3. Michael Sipser, Introduction to the Theory of Computation, PWS Publishing.
4. John Martin, Introduction to Languages and the Theory of Computation, Tata McGraw Hill.

List of Practical's:

1. Write a formal notation for strings, languages and machines.
2. Design finite automata to accept a set of strings of a language.
3. For a given language determine whether the given language is regular or not.
4. Design context free grammars to generate strings of context free language.
5. Determine equivalence of languages accepted by Push Down Automata and languages generated by context free grammars.
6. Write the hierarchy of formal languages, grammars and machines.
7. Distinguish between computability and non-computability and Decidability and undesirability.

PCC-CS 504	Object Oriented Programming	3L:0T:2P	4 Credits

Objectives: -

1. Understand fundamentals of programming such as variables, conditional and iterative execution, methods, etc.
2. Understand fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries, etc.
3. Be aware of the important topics and principles of software development.
4. Have the ability to write a computer program to solve specified problems.

Course Contents:-

Module 1 Introduction to OOP's **(04 Hrs)**

Need for OOP Paradigm, Procedural programming vs object oriented programming, object oriented concepts, Features.

Module 2 Functions: **(04 Hrs)**

Main function, function prototyping, inline functions, reference variables, call by reference, Defaults arguments, function overloading, and Math library functions.

Module 3 Classes & Objects: **(08 Hrs)**

Difference between C structure and class, specifying a class, Defining member functions: inside and outside class, scope resolution operator, Array within a class, array of objects, Static data members and member functions, Object as function arguments, returning objects, Friend function, memory allocation for objects, pointer to members, pointer to object, this pointer local classes.

Module 4 Constructor and Destructor: **(04 Hrs)**

Constructor, types of constructors: default, parameterized and copy constructor, constructor overloading, constructor with default parameter, dynamic initialization of objects, destructor.

Module 5 Inheritance and polymorphism: **(06 Hrs)**

Base class, derived class, visibility modes, derivation and friendship, Types of inheritance, Containership, virtual function binding, pure virtual function, Abstract class, pointer to derived class.

Module 6 Operator overloading and Type Conversion: **(06 Hrs)**

Defining operator overloading, overloading unary and binary operator, Data Conversion: Basic to User Defined, User defined to basic, Conversion from one user-defined to other.

Module 7 File Handling & IO operations - Console IO operations: **(04 Hrs)**

C++ stream classes, Unformatted IO operations, formatted IO operations, managing output with manipulators.

Module 8 Working with files: **(04 Hrs)**

Classes for file stream operations, opening and closing files, detectinf cof, File opening modes, file Pointers, Error handling during file operations, command line arguments. Templates: Class template, class template with parameter, function template, function template with parameter.

Text Books:

No single text book covers all of the topics in this course. For those new to OOP, the best introductions are usually found in the introductory programming texts for OOP languages (Java, python or C++).

1. **Object-Oriented Thought Process, The (3rd Edition):** Object-Oriented Thought Process, The (3rd Edition)
2. Thinking in C++, Vol. 1, 2e By, Bruce Eckel.

Reference Books:

1. Bjrane Stroustrup, "C++ Programming language",3rd edition, Pearson education Asia(1997)
2. Object Oriented Programming with C++,3rd edition McGraw Hill by E. Balagurusamy
3. Lafore R."Object oriented Programming in C++", 4th Ed. Techmedia, New Delhi(2002).
4. Yashwant Kenetkar,"Let us C++",1stEd.,Oxford University Press(2006)
5. B.A. Forouzan and R.F. Gilberg, CompilerScience,"A structured approach using C++" Cengage Learning, New Delhi.

List of Practical's:

1. Introduction to Object oriented Programming concepts.
2. Simple C++ Programs to Implement Various Control Structures.
   If statement
   Switch case statement and
3. Simple C++ Programs to Implement
   do while loop,For loop and While loops
4. Program to Understand Structure
5. Program to Understand Union
6. Programs to Understand Pointer Arithmetic using C++.
7. Functions & Recursion.
8. Program to Understand Inline Functions.

9. Programs to Understand Different Function Call Mechanism.
10. Programs to Understand Call by reference & Call by Value
11. Programs to Understand Storage Specifiers.
12. Programs for Constructors & Destructors.
13. Program to implement the copy constructor.
14. WAP to make use of "this" Pointer. Using class

    Programs to Implement Multiple inheritance –Access Specifiers
15. Program to Implement Function Overriding.

    Hierarchical inheritance – Function Overriding /Virtual Function
16. Programs to Overload Unary & Binary Operators as Member Function & Non Member Function.
17. Programs to Understand Friend Function & Friend Class.
18. WAP for Implement File Handling
19. WAP for copy the content of one file into other file using IO Stream classes
20. Study of mini project

Course Outcomes:-

1) To demonstrate the use of various oops concept with the help of program.
2) Describe the procedural and object oriented paradigm.
3) To demonstrate and implement real world problem.
4) Define classes modeling techniques and instances.
5) Design object oriented solutions for small systems involving multiple objects.
6) Apply virtual and pure virtual function & complex programming situation.

Elective-I

PEC-CS 505 (A)	Web and Internet	3L:0T:2P	4 Credits

Objectives of the course

1. To understand the various steps in Program development.
2. To understand the concepts of constants, variables ,datatypes & operators in C Programming Language.
3. To learn how to write algorithm and draw flowchart for a given problem statement of C Programs.
4. To learn to write programs (using structured programming approach) in C to solve problems.
5. To make the student understand simple sorting and searching methods.

The theory, practical experiences and relevant soft skills associated with this course are to be taught and implemented, so that the student demonstrates the following industry oriented CO's associated:

- Demonstrate the basic knowledge of computer hardware and software.
- Ability to apply solving and logical skills to programming in C language and also in other languages.
- Develop flowchart and algorithm to solve problems logically.
- This subject will act as a programming concept developer "for the students it will also act as a backbone" for other programming languages.

Course Contents:-

Module 1 Introduction to HTML: (10 Hrs)

Introduction to Networking, Internet and its evolution, Client/Server technology, Web page, web site, URI, Web Server, Web Client, Web Browser. Mark-up languages, Introduction to HTML, Elements and Attributes, Different sections of HTML document, comments, common tags for heading, paragraphs, Horizontal lines, line breaks, Formatting, Links, Images, Tables, Lists, Forms, Head, Meta and Div tags, Events**.**

Module 2 CSS: (08 Hrs)

Introduction to CSS, Syntax, Inserting CSS (External, Internal and Inline), ID and class selectors, grouping and nesting selectors, Pseudo classes and elements.

Module 3 XML: **(08 Hrs)**

Document type definition, XML Schemas, Document Object model, Presenting XML, Using XML Processors: DOM and SAX.

Module 4 DHTML & Client-Side Programming : **(10 Hrs)**

Introduction to Java script, writing comments, variables, operators, statements, alert, confirm and prompt boxes, functions, events and error handling, Introduction to built-in classes, form validation, cookies. Introduction to DHTML DHTML technologies (JavaScript, DOM, Events, CSS).

Reference Books:

1. "HTML5: Designing Rich Internet Applications" by Matthew David
2. HTML and CSS: Visual QuickStart Guide (8th Edition) (Visual Quickstart Guides)
3. Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics
4. HTML and CSS: Design and Build Websites
5. JavaScript: The Definitive Guide
6. Professional JavaScript for Web Developers

List of Practical's:

1. Know your laboratory
2. Introduction to HTML. Create a basic HTML file
3. Create a static webpage using table tags of HTML
4. Create a static web page which defines all text formatting tags of HTML in tabular format
5. Create webpage using list tags of HTML
6. Create webpage to include image using HTML tag
7. Create employee registration webpage using HTML form objects
8. Apply style sheet in Web page. [inline, embedded and linked]
9. Introduction to IIS. Installation of IIS server
10. Create a dynamic web page which displays a message "Welcome to Java Script" using JavaScript.
11. Create a dynamic web page which generates student grade sheet using JavaScript.
12. Create a dynamic web page which prints Fibonacci series from 1 to 10 in JavaScript
13. Create a dynamic web page which displays factorial of a number in JavaScript.

14. Create a dynamic web page which displays arithmetic operations [addition, subtraction, division, multiplication and modulus] using HTML Frame
15. Create a simple xml file and also create dynamic web page in which XML tags used
16. Display food menu using XML.
17. Design a simple form and validate for password and phone no using JavaScript.
18. Design a dynamic menu using DHTML.
19. Design a simple table with dynamic content using DHTML.

Course Outcomes:-

1) Understand and analyze role of scripting languages.
2) To aware about different web technologies.
3) To demonstrate content design for web.
4) Demonstrate procedure of client and server side scripting and validation.
5) To demonstrate use of web application software development tools.

Elective-I

PEC-CS 505 (B)	Graph Theory	3L:0T:2P	4 Credits

Objectives:-

1. Introduction to Graphs
2. Paths and Circuits
3. Trees and Fundamental circuits
4. Matrix representation of graphs
5. Planar and Dual graph
6. Colouring of planar graphs
7. Graph Algorithms

Course Contents:-

Module 1:- Introduction to Graphs: **(04 Hrs)**

Definition of a graph and directed graph, simple graph. Degree of a vertex, regular graph, bipartite graphs, sub graphs, complete graph, complement of a graph, operations of graphs, isomorphism and homomorphism between two graphs, directed graphs and relations.

Module 2 :-Paths and Circuits: **(06 Hrs)**

Walks, paths and circuits, connectedness of a graph, Disconnected graphs and their components, Konigsberg 7-bridge problem, Around the world problem, Euler graphs, Hamiltonian paths and circuits, Existence theorem for Eulerian and Hamiltonian graphs.

Moduel 3 Trees and Fundamental circuits: **(08 Hrs)**

Trees and their properties, distance and centre in a tree and in a graph, rooted and binary trees, spanning trees and forest, fundamental circuits, cut sets, connectivity and separability,1-isomorphism, 2-isomorphism, breadth first and depth first search.

Module 4:-Matrix representation of graphs: **(04 Hrs)**

Incidence matrix and its sub matrices, Reduced incidence matrix, circuit matrix, fundamental circuit matrix, cut set matrix, fundamental cut set matrix, path matrix, adjacency matrix of a graph and of digraph.

Moduel 5 Planar and Dual graph : **(08 Hrs)**

graphs, Euler's formula, Kuratowski's graphs, detections of planarity, geometric dual, combinatorial dual. Colouring of planar graphs: Chromatic number, independent set of vertices, maximal independent set, chromatic partitioning, dominating set, minimal dominating set, chromatic polynomial, coloring and four colour problem, coverings,

Module 6:-Graph Algorithms: (06 Hrs)

Network flows, Ford-Fulkerson algorithm for maximum flow, Dijkstra algorithm for shortest path between two vertices, Kruskal and Prim's algorithms for minimum spanning tree.

References:

1. Deo Narsingh, Graph Theory with Applications to engineering and computer science, Prentice Hall of India, 1992.

2. Clark John and Holton D.A., A first Look At Graph Theory, Allied Publishers Ltd., New Delhi,1995.

3. Aldous and Wilson, Graphs and Applications: An Introductory Approach, Springer,2000.

4. Mott J.L., Kandel A and Baker T.P., . Discrete Mathematics for Computer Scientists and Mathematicians, Prentice Hall of India,2001.

5. Reinhard Diestel, Graph Theory, Springer International Edition..2004

List of practical's:-.

1. Write a C Program to check whether an undirected graph contains Eulerian Cycle.
2. Write a C program to find the strongly connected component in a Graph.
3. Write a c program to check articulation point exist in a Graph or not.
4. Write a C Program to perform insertion in a binary tree.
5. Write a C Program to perform searching in binary search tree
6. Write a C Program to perform preorder, inorder and post order travels of a given binary tree
7. Write a C program to check the connectivity of Graph using BFS (Bread Firth Search)
8. Write a C program to check whether the given graph is planner or not.
9. Write a C program to find the transitive closure of given Graph G.
10. Write a C program to find Graph coloring for a given Graph G.
11. Write a C program to find the chromatic number of given planner Graph.
12. Write a C program to implement Ford-Fulkerson algorithm for maximum flow.
13. Write a C program to implement Dijkstra algorithm for shortest path between two vertices.
14. Write a C program to implement Kruskal algorithms for minimum spanning tree
15. Write a C program to implement Prim's algorithm for minimum spanning tree

Course Outcomes:-

1) To demonstrate and apply fundamental concept of graph theory.
2) To apply graph theory based tools in solving practical problems.
3) To improve the proof writing skill.
4) Analyze and apply graph theory on real world problems.

Elective-I

PEC-CS 505 (C)	Software Engineering	4L:0T:2P	4 Credits

Objectives:-

1. Understand the Software life cycle models.
2. Understand the importance of the Software development process.
3. Understand the importance of modeling and modeling languages.
4. Design and develop correct and robust software products.

Course Contents:-

Module 1 INTRODUCTION: **(8 Hrs)**

Software Engineering-Software Process- Generic process model-Prescriptive process model-specialized, unified process-Agile development-Agile Process-Extreme Programming- Other agile Process models-Software engineering Knowledge-core Principles-Principles that guide each framework Activity.

Module 2 REQUIREMENTS: **(8 Hrs)**

Requirements Engineering-Establishing the Groundwork-Eliciting Requirements- Developing use cases-Building the requirements model-Negotiating, validating Requirements-Requirements Analysis-Requirements Modeling Strategies.

Moduel 3 DESIGN: **(9 Hrs)**

MODELING WITH UML: Modeling Concepts and Diagrams - Use Case Diagrams- Class Diagrams - Interaction Diagrams - State chart Diagrams – Activity Diagrams - Package Diagrams - Component Diagrams – Deployment Diagrams -Diagram Organization- Diagram Extensions. Design Process- Design concepts: Abstraction, Architecture, patterns, Separation of Concerns, Modularity, Information Hiding, Functional Independence, Refinement, Aspects, Refactoring,

Object Oriented Design Concepts, Design Classes- Design Model: Data, Architectural, Interface, Component, Deployment Level Design Elements .

Module 4 SOFTWARE IMPLEMENTATION: **(8 Hrs)**

Structured coding Techniques-Coding Styles-Standards and Guidelines- Documentation Guidelines-Modern Programming Language Features: Type checking-User defined data types-Data Abstraction-Exception Handling-Concurrency Mechanism.

Module 5 TESTING AND MAINTENANCE: **(9 Hrs)**

TESTING: Software Quality- Software Quality Dilemma- Achieving Software Quality- Testing: Strategic Approach to software Testing- Strategic Issues-Testing: Strategies for Conventional Software, Object oriented software, Web Apps-Validating Testing- System Testing- Art of Debugging. MAINTENANCE: Software Maintenance-Software Supportability- Reengineering-Business Process Reengineering- Software Reengineering- Reverse Engineering-Restructuring-Forward Engineering- Economics of Reengineering

TEXT BOOKS

1. Roger S, "Software Engineering – A Practitioner's Approach", seventh edition, Pressman, 2010.

2. Pearson Edu, "Software Engineering by Ian sommerville", 9th edition, 2010.

REFERENCES

1. Hans Van Vliet, "Software Engineering: Principles and Practices"–, 2008.

2. Richard Fairley, "Software Engineering Concepts", 2008.

List of Practical's:

1. Draw the UML diagrams for a target system.
    a. Analysis diagrams: USE case view
    b. OO analysis: discovering classes
    c: Interaction diagrams: sequence and collaboration diagrams
    d: State Transition Diagram.
    e: Component and Deployment diagrams.
2. Design of test cases based on requirement and design.
3. Understand The Automation Testing Approach (Theory Concept).
4. Install Selenium server and demonstrate it using a script in Java/PHP.

5. Using Selenium IDE, Write a test suite containing minimum 4 test cases.

6. Manual Testing

1) Create Test Strategy
2) Create Test Plan
3) Create Test Sign Off
4) Create Requirement traceability matrix
5) Creating Test Cases
6) Take any online application e.g Amazon and cover all the test cases

7. Automation:

1) Setup Infrastructure for Selenium Testing
2) Installation and usage of TestNG,Jenkins,Maven,SVN/Git Hub,Java
3) Practical to create few Scripts for any online application
4) Check how to Create Report e,g Extent report, Allure, XSLT
5) TestNG Annotation with execution of few suites
6) Schedule job using Jenkins
7) Creating version control using GIT HUB Sending Automatic execution email using Jenkins

8. Conduct a test suite for nay two web sites.

9. Write and test a program to login a specific web page.

10. Write and test a program to update 10 student records into table into Excel file.

11. Write and test a program to select the number of students who have scored more than 60 in any one subject.

12. Write and test a program to provide total number of objects present available on the page.

13. Write and test a program to get the number of list items in a list combo box.

Course Outcomes:-

1) Define various software application domains and process models used in software development.
2) Need of software specification and requirement and their gathering techniques.
3) Demonstrate use of software and user interface design principles.
4) Classify different testing strategies and statistics and compare them.
5) Generate project schedule and organize different activities of project as per risk impact factor.

MC 506	Constitution of India/Essence of Indian Traditional Knowledge	2L:0T:0P	0 Credits

Module1 Introduction Constitution:- (04 Hrs)

Indian Constitution: Sources and constitutional history, Features: Citizenship, Preamble, Fundamental Rights and Duties, Directive Principles of State Policy

Module 2 Union Government and its Administration:- (04 Hrs)

Structure of the Indian Union: Federalism, Centre- State relationship, President: Role, power and position, PM and Council of ministers, Cabinet and Central Secretariat, Lok Sabha, Rajya Sabha

Module 3 State Government and its Administration:- (04 Hrs)

Governor: Role and Position, CM and Council of ministers, State Secretariat: Organization,

Structure and Functions

Module 4 Local Administration:- (06 Hrs)

District's Administration head: Role and Importance,

Municipalities: Introduction, Mayor and role of Elected Representative, CEO of Municipal

Corporation, Pachayati raj: Introduction, PRI: Zila Pachayat, Elected officials and their roles,

CEO Zila Pachayat: Position and role, Block level: Organizational Hierarchy (Different departments), Village level: Role of Elected and Appointed officials, Importance of grass root democracy

Module 5 Election Commission: (06 Hrs)

Election Commission: Role and Functioning, Chief Election Commissioner and Election Commissioners, State Election Commission: Role and Functioning, Institute and Bodies for the welfare of SC/ST/OBC and women

Module 6 Astronomy, Chemistry, Mathematics and Metallurgy in India:– (06 Hrs)

Siddhantik and Post siddhantik development of Astronomy, Early Chemical Techniques, Atomism in Vaiśeṣika, Chemistry in Early Literature,First Steps, Early Historical Period, The Classical Period, The Classical Period, post-Āryabhaṭa, The Kerala School of Mathematics, Features of Indian MathematicsMetallurgy before and during the Harappan Civilization, After the Harappans, iron metallurgy, wootz steel, other iron pillars and beams, zinc, social context

Module 7 Medical Sciences in India: **(04 Hrs)**

The Principles of Ayurvedic Healing, Treating diseases to restore health.

Module 8 Music, Theater and Drama in India – (04 Hrs)

Origin, classification accompanied instrument, Bharata's Nāṭyaśāstra, New era, Medieval period, modern era, aesthetics of Indian classical music, forms of composition: Dhrupada, thumari, gazal, tarana, tappa, folk music, film musicTheater and Dram Its Beginnings, Classical Period, Major Indian Dramatists: Bhāsa, Kālidāsa, Bhavabhūti, Medieval Period, Kuṭiyaṭṭam, Yakṣagāna, Bhavāī, Jātrā, Nautaṅkī, Swāṅg, Rāmalīlā, Tamāśā, Nāchā, Pāṇḍavānī, Modern Era

References

1. 'Indian Polity' by Laxmikanth
2. 'Indian Administration' by Subhash Kashyap
3. 'Indian Constitution' by D.D. Basu
4. Āryabhaṭa
5. Vātsyāyana, Nāgārjuna, Al-Bīrūnī', Vāgbhaṭa
6. Taittīriya Brāhmaṇa, Yājñavalkya Smṛti, Viṣṇu Purāṇa, Skanda Purāṇa,
7. Nātyaśāstra, Viṣṇudharmottarapurāṇa-Khaṇḍa III, Bhakti Movement

HSMC 507	Seminar-II	0L:0T:2P	1 Credits

This seminar is based on the recent advances in Computer Science. Student has to write a paper in IEEE format on any recent topic pertaining to Computer science by referring different journals. Student has to prepare PPT of the same and present before students and faculties.

Course Name : Third Year Computer Engineering

Semester : Fifth

Subject Title : NPTEL Course - I

Subject Code : NPTEL-508

Teaching Scheme (in hrs)			Total Credit (TH +T+P)	Examination Scheme					
				Theory		Practical			Total
TH	T	P	CR	MSE	ESE	PR	OR	TW	
--	--	02	01	--	--	--	--	--	--

Students have to complete minimum four weeks NPTEL web and video course from Computer Engineering Department which is available on portal nptel.ac.in. It is preferred that student should attend any one course related to subjects of Fifth semester.

Certification courses are offered twice a year (Jan-Jun, Jul-Dec). Joining a course is free. Learning can be done by watching videos and this is tested by the weekly assignments, which are to be submitted online within the prescribed deadline.

There is a certification examination that the student can take for a nominal fee at the end of the course to earn certificates from the IITs.

To earn credits of this course students have to produce the NPTEL course completion certificate and online submitted assignments to the department before end semester practical examination.

Suggested Courses:

1. **Object Oriented programming (week 12)**
2. **Software engineering (week 8)**
3. **Database management system (week 8)**
4. **Software testing (week 12)**
5. **Ethical hacking (Week 12)**

PCC-CS 601	Compiler Designing	3L:0T:2P	4 Credits

OBJECTIVES:

The student should be made to:

1. Learn the design principles of a Compiler.
2. Learn the various parsing techniques and different levels of translation
3. Learn how to optimize and effectively generate machine codes

Course contents:

Module 1 INTRODUCTION TO COMPILERS :- (04 Hrs)

Translators-Compilation and Interpretation-Language processors -The Phases of Compiler-Errors Encountered in Different Phases-The Grouping of Phases-Compiler Construction Tools - Programming Language basics.

Module 2 LEXICAL ANALYSIS:– (06 Hrs)

Need and Role of Lexical Analyzer-Lexical Errors-Expressing Tokens by Regular Expressions-Converting Regular Expression to DFA-Minimization of DFA-Language for Specifying Lexical Analyzers-LEX-Design of Lexical Analyzer for a sample Language.

Module 3 SYNTAX ANALYSIS : (08 Hrs)

Need and Role of the Parser-Context Free Grammars -Top Down Parsing -General Strategies-Recursive Descent Parser Predictive Parser-LL(1) Parser-Shift Reduce Parser-LR Parser-LR (0)Item- Construction of SLR Parsing Table -Introduction to LALR Parser - Error Handling and Recovery in Syntax Analyzer-YACC-Design of a syntax Analyzer for a Sample Language .

Module 4 SYNTAX DIRECTED TRANSLATION : (06 Hrs)

Syntax directed Definitions-Construction of Syntax Tree-Bottom-up Evaluation of S-Attribute Definitions- Design of predictive translator - Type Systems-Specification of a simple type checker- Equivalence of Type Expressions-Type Conversions.

Module 5 RUN-TIME ENVIRONMENT: (06 Hrs)

Source Language Issues-Storage Organization-Storage Allocation-Parameter Passing-Symbol Tables-Dynamic Storage Allocation-Storage Allocation in FORTAN.

Module 6 CODE OPTIMIZATION AND CODE GENERATION: (06 Hrs)

Principal Sources of Optimization-DAG- Optimization of Basic Blocks-Global Data Flow Analysis- Efficient Data Flow Algorithms-Issues in Design of a Code Generator - A Simple Code Generator Algorithm.

TEXTBOOK:

1. Alfred V Aho, Monica S. Lam, Ravi Sethi and Jeffrey D Ullman, "Compilers – Principles, Techniques and Tools", 2nd Edition, Pearson Education, 2007.

REFERENCES:

1. Randy Allen, Ken Kennedy, "Optimizing Compilers for Modern Architectures: A Dependence-based Approach", Morgan Kaufmann Publishers, 2002.
2. Steven S. Muchnick, "Advanced Compiler Design and Implementation, "Morgan Kaufmann Publishers - Elsevier Science, India, Indian Reprint 2003.
3. Keith D Cooper and Linda Torczon, "Engineering a Compiler", Morgan Kaufmann Publishers Elsevier Science, 2004.
4. Charles N. Fischer, Richard. J. LeBlanc, "Crafting a Compiler with C", Pearson Education, 2008.

List of Practical's:-

1. For a given grammar specification develop the lexical analyser
2. For a given parser specification design top-down and bottom-up parsers
3. Develop syntax directed translation schemes
4. Develop algorithms to generate code for a target machine
5. Design a lexical analyzer for given language and the lexical analyzer should ignore redundant spaces, tabs and new lines. It should also ignore comments. Although the syntax specification states that identifiers can be arbitrarily long, you may restrict the length to some reasonable value. Simulate the same in C language.
6. Write a C program to identify whether a given line is a comment or not
7. Write a C program to recognize strings under 'a', 'a*b+', 'abb'.
8. Write a C program to test whether a given identifier is valid or not
9. Write a C program to simulate lexical analyzer for validating operators. Implement the lexical analyser using JLex, flex or other lexical analyzer generating tools.
10. Write a C program for Implementing the functionalities of predictive parser for the mini language specified in Note 1
11. a). * Write a C program for constructing of LL (1) parsing
    b). *Write a C program for constructing recursive descent parsing
12. Write a C program to implement LALR parsing.

13. Convert the BNF rules into Yacc form and write code to generate abstract syntax tree for the mini language specified in Note 1
14. Write a C program to generate machine code from abstract syntax tree generated by the parser. The instruction set specified in Note 2 may be considered as the target code.

PCC-CS 602	Computer Networks	3L:0T:2P	4 Credits

Objectives of the course

1. To develop an understanding of modern network architectures from a design and performance perspective.
2. To introduce the student to the major concepts involved in wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs).
3. To provide an opportunity to do network programming
4. To provide a WLAN measurement ideas.

Course contents

Module 1: (04 Hrs)

Data communication Components: Representation of data and its flow Networks , Various Connection Topology, Protocols and Standards, OSI model, Transmission Media, LAN: Wired LAN, Wireless LANs, Connecting LAN and Virtual LAN, Techniques for Bandwidth utilization

Module 2: Multiplexing: (04 Hrs)

Frequency division, Time division and Wave division, Concepts on spread spectrum.

Module 3 Data Link Layer: (08 Hrs)

Error Detection and Error Correction - Fundamentals, Block coding, Hamming Distance, CRC; Flow Control and Error control protocols Stop and Wait, Go back – N ARQ, Selective Repeat ARQ, Sliding Window, Piggybacking, Random Access, Multiple access protocols - Pure ALOHA, Slotted ALOHA, CSMA/CD,CDMA/CA

Module 4 Network Layer: (08 Hrs)

Switching, Logical addressing – IPV4, IPV6; Address mapping – ARP, RARP, BOOTP and DHCP–Delivery, Forwarding and Unicast Routing protocols.

Module 5 Transport Layer: ((06 Hrs)

Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), SCTP Congestion Control; Quality of Service, QoS improving techniques: Leaky Bucket and Token Bucket algorithm.

Module 6 Application Layer: (06 Hrs)

Domain Name Space (DNS), DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls, Basic concepts of Cryptography

Suggested books:

1. Data Communication and Networking, 4th Edition, Behrouz A. Forouzan, McGraw-Hill.
2. Data and Computer Communication, 8th Edition, William Stallings, Pearson Prentice Hall India.

Suggested reference books

1. Computer Networks, 8th Edition, Andrew S. Tanenbaum, Pearson New International Edition.
2. Internetworking with TCP/IP, Volume 1, 6th Edition Douglas Comer, Prentice Hall of India.
3. TCP/IP Illustrated, Volume 1, W. Richard Stevens, and Addison-Wesley, United States of America.

List of Practical's:

1. Observe of network laboratory components. Write specifications of latest desktops and laptops.
2. Observe, identify and understand different Transmission Media and Network Control devices.
3. Study of different types of Network cables and Practically implement the cross-wired cable and straight through cable using clamping tool. Use LAN Tester to check cable.
4. Install a network interface card
5. Study of Network Devices in Detail.
6. Connect Computers in Star Topology using Wired Media and any Network control Device.
7. Connect two hubs/switch by creating crossover connection
8. Study of network IP.
9. Study of router and other internetworking device.
10. Configure Peer-to-Peer Network.
11. Connect the computers in Local Area Network.
12. Configure a Network topology using packet tracer software.
13. Share Printer and Folder in Network.
14. Configure advanced features of TCP/IP Protocols.
15. Install Wireshark software to capture packet and configure it to capture Ethernet packet. Verify Ethernet frame structure and its 48 bit address.

16. To Run Basic TCP/IP Utilities and Network Commands with all options.(Ping, Ping ::1, ipconfig, Tracert, Netstat, Wireshark, ARP, NBTSTAT.EXE, WINIPCFG.EXE),
17. Use Wireshark packet sniffer software to capture TCP, UDP, IP, ARP, ICMP, Telnet, FTP packets
18. Designing and implementing Class A, B, and C Networks.
19. Subnet planning and its implementation.
20. Study Subnet Masking and create two subnets
21. Installation of ftp server and client

Elective-II

PEC 603 (A)	Advanced Computer Architecture	3L:0T:2P	4 Credits

Course Outcomes:

At the end of this course, students will demonstrate the ability to

1. Understand the concepts of microprocessors, their principles and practices.
2. Write efficient programs in assembly language of the 8086 family of microprocessors.
3. Organize a modern computer system and be able to relate it to real examples. Develop the programs in assembly language for 80286, 80386 and MIPS processors in real and protected modes.
4. Implement embedded applications using ATOM processor.

Course Contents:-

Module 1: Introduction to computer organization:- (06 Hrs)

Architecture and function of general computer system, CISC Vs. RISC, Data types, Integer Arithmetic - Multiplication, Division, Fixed and Floating point representation and arithmetic, Control unit operation, Hardware implementation of CPU with Micro instruction, microprogramming, System buses, Multi-bus organization.

Module 2: Memory organization:- (05 Hrs)

System memory, Cache memory - types and organization, Virtual memory and its implementation, Memory management unit, Magnetic Hard disks, Optical Disks.

Module 3: Input – output Organization – (05 Hrs)

Accessing I/O devices, Direct Memory Access and DMA controller, Interrupts and Interrupt Controllers, Arbitration, Multilevel Bus Architecture, Interface circuits - Parallel and serial port. Features of PCI and PCI Express bus.

Module 4: 16 and 32 microprocessors- (08 Hrs)

80x86 Architecture, IA – 32 and IA – 64, Programming model, Concurrent operation of EU and BIU, Real mode addressing, Segmentation, Addressing modes of 80x86, Instruction set of 80x86, I/O addressing in 80x86

Module 5: Pipelining- (06 Hrs)

Introduction to pipelining, Instruction level pipelining (ILP), compiler techniques for ILP, Data hazards, Dynamic scheduling, Dependability, Branch cost, Branch Prediction, Influence on instruction set.

Module 6: Different Architectures – (06 Hrs)

VLIW Architecture, DSP Architecture, SoC architecture, MIPS Processor and programming

Text/Reference Books

1. V. Carl, G. Zvonko and S. G. Zaky, **"**Computer organization**"**, McGraw Hill, 1978.
2. B. Brey and C. R. Sarma, **"**The Intel microprocessors**"**, Pearson Education, 2000.
3. J. L. Hennessy and D. A. Patterson, **"**Computer Architecture A Quantitative Approach**"**, Morgan Kauffman, 2011.
4. W. Stallings, **"**Computer organization**"**, PHI, 1987.
5. P. Barry and P. Crowley, **"**Modern Embedded Computing**"**, Morgan Kaufmann, 2012.
6. N. Mathivanan, **"**Microprocessors, PC Hardware and Interfacing**"**, Prentice Hall, 2004.
7. Y. C. Lieu and G. A. Gibson, **"**Microcomputer Systems: The 8086/8088 Family**"**, Prentice Hall India, 1986.
8. J. Uffenbeck, **"**The 8086/8088 Design, Programming, Interfacing**"**, Prentice Hall, 1987.
9. B. Govindarajalu, **"**IBM PC and Clones**"**, Tata McGraw Hill, 1991.
10. P. Able, **"**8086 Assembly Language Programming**"**, Prentice Hall India.

List of Practical's:-

1. Give the Presentation on Von Neumann Architecture of a computer system .

2. Give the presentation on Memory Management virtual memory cache memory and paging.

3. Familiarity with state of art IC-chips, e.g. a) Multiplexer , b) Decoder, c) Encoder, d) Counter, e)Shift-Register, f)adder Truth Table verification and clarification from Data-book.

4. Design a BCD adder.

5. Design a carry-look ahead Adder.

6. Design a ripple counter and carry-look ahead counter and assess the complexity of both the ckts.

7. Use a multiplexer unit to design a composite ALU .

8. Design a multiplex display unit using counter, multiplexer, decoder etc.

9. Design a keyboard Encoder unit in 2 Dimension.

10. Test a RAM chip and cascade two chips for vertical and horizontal expansion. Use wired OR tri-state output interconnection.

11. Use ALU chip for multibit arithmetic operation.

12. Write an assembly level program for the following pseudocode

SUM = 0

SUM = SUM + A + B

DIF = DIF –C

SUM = SUM + DIF.

Elective-II

PEC 603 (B)	Advanced Operating Systems	3L:0T:2P	4 Credits

Objectives

1. Define, explain, and apply operating systems concepts: process management, CPU scheduling, synchronization, memory management, file system, and the like.
2. Use the operating system interface.
3. Gain experience in implementing and debugging operating system components, including the kernel module, system call, synchronization primitives, and the file system.

Course Contents:-

Module 1 Introduction to distributed operating Systems: (06 Hrs)

Advanced topics in concurrency, deadlock protection, scheduling, computer system modelling, virtual memory management, distributed and interposes communication, distributed transactions and computations, and distributed system design.

Module 2 Distributed file systems: (06 Hrs)

Introduction, interconnections, distributed system taxonomy, service models, client-server computing, network protocols, Coda, DFS, SMB/CIFS, Google FS (GFS), GmailFS, xFS

Module 3 Communication in distributed systems: (08 Hrs)

Distributed mutual exclusion algorithms, Performance matrix, Message passing communication, Remote procedure call, Transaction communication Group, communication, Broadcast atomic protocols

Module 4: Introduction to Memory Management: (08 Hrs)

What is Memory & Why Manage It? Process Compilation & Memory Locations ,Bare Machine, Operating System in ROM – Resident Monitor, Partitions, Allocating & Placing Partitions in Memory, Problems with Partitions, Pages ,An Example Page Entry ,Pages vs. Paging ,Huge Logical Memory Maps ,Problems of Paged Memory Management ,Sharing Pages ,Copy-on-Write ,Operating System Use of Page Entry Protections

Module 5: Virtual Memory: (08 Hrs)

Why Use Virtual Memory? ,Paging ,Paging – How It Works ,Hardware Requirements ,Optimal Page Replacement ,Not Recently Used Algorithm ,FIFO Algorithm – First In, First Out ,Second Chance ,Least Recently Used ,VM Problems , Initial Process Memory Allocation.

Reference Books:-

1. "Design of the UNIX Operating System: United States Edition (Prentice Hall Software Series)" by Maurice J Bach.
2. "The Design and Implementation of the FreeBSD Operating System" by Marshall Kirk McKusick and George V Neville-Neil
3. "Operating Systems Design and Implementation (Prentice Hall Software Series)" by Andrew S Tannenbaum
4. "Real-Time Systems" by Jane W S Liu
5. "Open Distributed Systems: On Concepts, Methods, and Design from a Logical Point of View (Vieweg Advanced Studies in Computer Science)" by Reinhard Gotzhein
6. "Distributed Operating Systems: Concepts and Design" by Sinha
7. "Operating Systems: Internals and Design Principles" by Stallings
8. "Operating Systems Design and Implement at: Design and Implementation" by Tanenbaum /Woodhull

List of Practical's:-

1. Basic LINUX commands and its Use.
2. Study of editors in LINUX
3. Detail study of File Access Permission in LINUX.
4. Detail study of LINUX Shell Programming.
5. Advance Shell Programming.
6. Programs on UNIX System calls.
7. Programs on process creation and synchronization, inter process communication including shared memory, pipes and messages.(Dinning Philosopher problem / Cigarette Smoker problem / Sleeping barber problem).
8. Simulation of CPU Scheduling Algorithms. (FCFS, RR, SJF, Priority, Multilevel Queuing).
9. Simulation of Banker's Algorithm for Deadlock Avoidance.
10. Program for FIFO, LRU, and OPTIMAL page replacement algorithm.

Elective-III

PEC 604 (A)	Artificial Intelligence	3L:0T:2P	4 Credits

Objectives:-

Students completing this course will have an in-depth understanding of three core areas of AI and the connections among them, and with such other key AI areas as machine learning, robotics, natural language processing and multi-agent systems. They should be able to:

1. Choose the appropriate representation for an AI problem or domain model, and construct domain models in that representation
2. Choose the appropriate algorithm for reasoning within an AI problem domain
3. Implement and debug core AI algorithms in a clean and structured manner
4. Design and analyze the performance of an AI system or component
5. Describe AI algorithms and representations and explain their performance, in writing and orally
6. Critically read papers on AI systems.

Course Contents:-

Module 1 Introduction: (04 Hrs)

The AI problems, AI technique, philosophy and development of Artificial intelligence.

Module 2 Problem Spaces and Search: (06 Hrs)

State space search, Uninformed and informed search techniques: BFS, A*, variations of A*. Local search and optimization: hill climbing, simulated annealing.

Module 3 Adversarial Search and Game Playing: (06 Hrs)

Minimax algorithm, alpha-beta pruning, stochastic games, Constraint satisfaction problems.

Module 4 Knowledge and Reasoning: (08 Hrs)

Logical agents, Propositional logic, First-order logic, Inference in FoL: forward chaining, backward chaining, resolution, and Knowledge representation: Frames, Ontologies, Semantic web and RDF.

Module 5 Introduction to PROLOG: (08 Hrs)

Facts and predicates, data types, goal finding, backtracking, simple object, compound objects, use of cut and fail predicates, recursion, lists, and simple input/output, dynamic database.

Module 6 Uncertain knowledge and reasoning: (06 Hrs)

Probabilistic reasoning, Bayesian networks, Fuzzy logic.

Reference Books-

1. "Artificial Intelligence: A Modern Approach" by Stuart Russell and Peter Norvig
2. "Artificial Intelligence: A New Sythesis" by Nils J Nilsson
3. "Artificial Intelligence" by Negnevitsky
4. "Artificial Intelligence and Machine Learning" by Anand Hareendran S and Vinod Chandra S S "Artificial Intelligence Techniques for Computer Graphics illustrated edition" by Plemenos

List of Practical's:-

1. Study of facts, objects, predicates and variables in PROLOG.
2. Study of Rules and Unification in PROLOG.
3. Study of "cut" and "fail" predicate in PROLOG.
4. Study of arithmetic operators, simple input/output and compound goals in PROLOG.
5. Study of recursion in PROLOG.
6. Study of Lists in PROLOG.
7. Study of dynamic database in PROLOG.
8. Study of string operations in PROLOG. Implement string operations like substring, string position, palindrome etc.)
9. Write a prolog program to maintain family tree.
10. Write a prolog program to implement all set operations (Union, intersection, complement etc.)
11. Write a prolog program to implement Library Management system.
12. Write a prolog program to solve "Water Jug Problem".

Elective-III

PEC 604 (B)	Data Mining	3L:0T:2P	4 Credits

Objectives:-

1. To introduce students to the basic concepts and techniques of Data Mining.
2. To develop skills of using recent data mining software for solving practical problems.
3. To gain experience of doing independent study and research.

Course Contents:-

Module 1 Introduction to Data Mining: (04 Hrs)

What is data mining?, Related technologies - Machine Learning, DBMS, OLAP, Statistics, Data Mining Goals, Stages of the Data Mining Process, Data Mining Techniques, Knowledge Representation Methods, Applications, Example: weather data

Module 2 Data Warehouse and OLAP: (04 Hrs)

Data Warehouse and DBMS, Multidimensional data model, OLAP operations, Example: loan data set

Module 3 Data pre-processing: (06 Hrs)

Data cleaning, Data transformation, Data reduction, Discretization and generating concept hierarchies, Installing Weka 3 Data Mining System, Experiments with Weka - filters, discretization

Module 4 Data mining knowledge representation: (06 Hrs)

Task relevant data, Background knowledge, Interestingness measures, Representing input data and output knowledge, Visualization techniques, Experiments with Weka - visualization

Module 5 Attribute-oriented analysis: (06 Hrs)

Attribute generalization, Attribute relevance- Class comparison, Statistical measures, Experiments with Weka - using filters and statistics

Module 6 Data mining algorithms: (06 Hrs)

Association rules, Motivation and terminology, Example: mining weather data, Basic idea: item sets, Generating item sets and rules efficiently, Correlation analysis, Experiments with Weka - mining association rules

Module 7 Data mining algorithms: **(06 Hrs)**

Classification Basic learning/mining tasks, Inferring rudimentary rules: 1R algorithm, Decision trees, Covering rules, Experiments with Weka - decision trees, rules, Data mining algorithms: Prediction, The prediction task, Statistical (Bayesian) classification, Bayesian networks, Instance-based methods (nearest neighbour), Linear models, Experiments with Weka – Prediction, Evaluating what's been learned, Basic

issues, Training and testing, Estimating classifier accuracy (holdout, cross-validation, leave-one-out).

References:

1. W. H. Inmon, "Building the Data Warehouse", 3rd edition..
2. Anahory and Murray., Data warehousing in the real world , Pearson Education/Addison Wesley.
3. Margaret Dunham, Data Mining: Introductory and Advanced Topics, Published by Prentice Hall.
4. Jiawei Han, Micheline Kamber, "Data Mining: Concepts and Techniques", Morgan Kaufmann Publishers, 2002.(www.cs.sfu.ca/~han/DMbook.html)
5. Alex Berson, Stephen J. Smith, "Data Warehousing, Data Mining, & OLAP", Tata Mcgraw- Hill, 2004.
6. George M Marakas, Modern Data Warehousing, Mining and Visualization-, Peason Education.

List of Practical's:-

1. Demonstration of preprocessing on dataset student.arff
2. Demonstration of preprocessing on dataset labor.arff
3. Demonstration of Association rule process on dataset contactlenses.arff using apriori algorithm
4. Demonstration of Association rule process on dataset test.arff using apriori algorithm
5. Demonstration of classification rule process on dataset student.arff using j48 algorithm
6. Demonstration of classification rule process on dataset employee.arff using j48 algorithm
7. Demonstration of classification rule process on dataset employee.arff using id3 algorithm
8. Demonstration of classification rule process on dataset employee.arff using naïve bayes algorithm
9. Demonstration of clustering rule process on dataset iris.arff using simple k-means
10. Demonstration of clustering rule process on dataset student.arff using simple k-means

Open Elective-I

OEC 605 (A)	(Humanities)-Soft Skills an interpersonal communication	3L:0T:0P	3 Credits

Module 1 Soft Skills: (04 Hrs)

What are soft skills? – Importance of soft skills , Attributes regarded as soft skills – Soft skills – Social - Soft skills – Thinking - Soft skills – Negotiating – Exhibiting your soft skills – Identifying your soft skills – Improving your soft skills –Exercise : Measure your soft skills

Module 2 Self-Discovery Importance of knowing yourself: (04 Hrs)

Process of knowing yourself - SWOT analysis - Benefits of SWOT analysis - Using SWOT analysis - SWOT analysis

Module 3 Developing Positive Attitude Meaning of attitude: (06 Hrs)

Features of attitudes - Attitude and behaviour - Formation of attitudes - Change of attitudes - What can you do to change attitude? - Ways of changing attitude in a person - Attitude in a workplace - The power of positive attitude - Developing positive attitude - Obstacles in developing positive attitude - Staying positive - Examples of positive attitudes - Positive attitude and its results - Staying negative - Examples of negative attitude - Overcoming negative attitude - Negative attitude and its results. Exercise: Measure your attitude

Module 4 Forming Values: (03 Hrs)

Meaning of value, a core of values - Values relating to education - Values relating to self and others - Values relating to civic

Module 5 Improving Perception: (03 Hrs)

Factors influencing perception - Perceptual process - Improving perception - Perception and its application in organizations. Exercise: Test your perception

Module 6 Body Language Body talk: (06 Hrs)

Voluntary and involuntary body language - Forms of body language - Parts of body language - Origin of body language - Uses of body language - Body language in building interpersonal relations - Body language in building industrial relations - Reasons to study body language - Improving your body language - Types of body language - Gender differences - Female interest and body

Module 7 Team Building and Teamwork: (04 Hrs)

Aspects of team building - Skills needed for teamwork, A model of team building - Team Vs Group - Characteristics of effective team - Role of a team leader - Role of team members - Nine persons a successful team should have - Inter-group collaboration - Advantages of inter-group collaboration

Module 8 Time Management : **(03 Hrs)**

Examination of Work, Sense of Time Management, Features of Time, Time Management Matrix, Difficulties in time management, Ideal way of spending a day

Module 9 Stress Management: **(03 Hrs)**

Meaning of stress, type of stress, effect of stress, Sources of stress, identifying existence of stress, Sign of stress and stress management.

Reference Books

1. Soft Skills – Know yourself and Know your world by Dr.K.Alex – S.Chand and Publications, New Delhi
2. Presonality development and soft skills –by Barun K Mishra – Oxford University Press.-2011

Open Elective-I

OEC 605 (B)	(Humanities)-ICT for development	3L:0T:0P	3 Credits

Course objectives:

With rising use of Information and Communication technologies available, there is a high potential for these technologies to address sustainability issues.

1. The students of MA (SDP), as development practitioners must be equipped with the knowledge about their applications in the development field so as to enable them to provide ICT solutions to the target communities.
2. This is an elective course intended for students who would like to gain knowledge and skills on how ICTs can be best used to overcome sustainability challenges.
3. In order to succeed in the practice of sustainable development, professionals must be trained in a basic set of competencies that integrate cross-disciplinary knowledge for practical problem solving with the use of information and communication technologies.

Course Content

Module 1:- **(04 hrs)**

Introduction to ICTs for sustainable Development Introduction to Information and Communication Technology (ICT); Role of ICTs in Sustainable Development; Current Statusof ICTs in Sustainable Development-Global and India Scenario. Potential of ICTs in various fields, impact of information Technologies on GDP growth.

Module 2:- **(06 hrs)**

Building Knowledge Societies-The concept of Knowledge Society; identifying stakeholders and target communities; Understanding information needs ,Traditional vs. contemporary knowledge systems, information processing and retrieval; Understanding means of communication in different areas, developing an effective communication strategy Case: Warna Unwired.

Module 3:- **(06 hrs)**

Information and Communication Technologies The hardware and software, the physical infrastructure, satellite, wireless solutions, telecommunication technologies, mobiles, fixed line, internet and world wide web, community radio, technology-user interface, design of relevant ICT products and services.

Module 4:- **(06 hrs)**

ICT Applications Applications of ICTin education, Health (telehealth, telemedicine and health informatics), Gender Equality, Agriculture (, e Governance, telecentres, Mobiles for

development, climate change and disaster management, ICT Networks for water management (This module will bedealt with the help of country case studies in all the sectors and inputs from ICT4D practitioners Case Studies: eCME, Apollo Telemedicine Network Foundation, Bhoomi, eSewa, Gyandoot, eAgriculture. M-ESA, CYCLETEL).

Module 5:- **(08 hrs)**

ICT for Development In India Policy and Institutional Framework in India, e governance, ICTModels in health, education , agriculture, finance, gender equality, Mobiles for Development Experience sharing by ICT for Development practitioners Case Studies: Reuters Market Light, Iffco Kisaan Sanchar Ltd.

Module 6:- **(06 hrs)**

ICT4D Implementation Developing an ICT4D Project, Critical Success factors for technology diffusion and use, Constraints in adoption, The role of national policies, Institutional Policy framework, Multi-stakeholder partnerships, Role of Private Sector Case Studies: echaupal , Lifelines India

PROJ-CS 606	Project-I (Mini Project)	0L:0T:4P	2 Credits

Students has to make extensive literature survey and identify the problem in view to complete Final Project of Eighth semester. Problem identification and pre work of the project should be carried out and presented in the department.

HSMC 607	Seminar-III	0L:0T:2P	1 Credits

This seminar is based on the recent advances in Computer Science. Student has to write a paper in IEEE format on any recent topic pertaining to Computer science by referring different journals. Student has to prepare PPT of the same and present before students and faculties.

HSMC 608	Technical & Competetive Skills	0L:0T:2P	1 Credits

Module 1 Gate Exam Preparation:

Orientation of GATE Curriculum for Mechanical Engineering, Providing information regarding literature of GATE Examination. Solving some sample question papers of GATE Examination. Giving information for Use of GATE for Job in PSU, Direct recruitment to Group A level posts in Central government and state Government.

Module 2:

Preparation of Engineering Mathematics and General Aptitude (GA) with Language and Analytical Skills for GATE examination.

Module 3:

Information regarding IES Examination and Recruitment procedure of Graduate Engineering students with detail curriculum, Literature and Guidance.

Module 4:

Information regarding Technical MPSC Examination and Recruitment procedure of Graduate Engineering students with detail curriculum, Literature and Guidance.

Module 5:

Technical Post, Curriculum and authentic literature of RRB, BSRB examination

Module 6:

Information Regarding Higher Education in Foreign Universities, Preparation of Pre requirements like SAT,PTE, LSAT,ACT, CAE,CPE GMAT, GRE, IELTS and the TOEFL.

Module 7:

Preparation for PG entrance examination – Curriculum and information of entrance examination to IIM and other MBA collages.

Module 8:

Information regarding different Scholarship offered For Higher Studies abroad to the Indian students.

Semester : Sixth

Subject Title : NPTEL Course - II

Subject Code : NPTEL-CSE609

Teaching Scheme (in hrs)			Total Credit (TH +T+P)	Examination Scheme					
				Theory		Practical			Total
TH	T	P	CR	MSE	ESE	PR	OR	TW	
--	--	02	01	--	--	--	--	--	--

Students have to complete minimum four weeks NPTEL web and video course from Computer Engineering Department which is available on portal nptel.ac.in. It is preferred that student should attend any one course related to subjects of Sixth semester.

Certification courses are offered twice a year (Jan-Jun, Jul-Dec). Joining a course is free. Learning can be done by watching videos and this is tested by the weekly assignments, which are to be submitted online within the prescribed deadline.

There is a certification examination that the student can take for a nominal fee at the end of the course to earn certificates from the IITs.

To earn credits of this course students have to produce the NPTEL course completion certificate and online submitted assignments to the department before end semester practical examination.

Suggested Courses:

1. Introduction to AI (week 12)
2. Data mining (week 8)
3. Advanced computer architecture (week 8)
4. Computer Network (week 12)
5. Python for data science (week 4)