ARULMIGU PALANIANDAVAR ARTS COLLEGE FOR WOMEN

(AUTONOMOUS)

Re-accredited with B^{++} by NAAC in 3rd cycle

(Run by Hindu Religious and Charitable Board under the Aegis of Arulmigu Dhandayuthapani Swamy Thirukovil, Palani)

(Affiliated to Mother Teresa Women's University, Kodaikanal)

Chinnakalayamputhur (PO), Palani 624 615.

Curriculum Framework and Syllabus for Outcome Based Education in

Master of Science
(Computer Science)

(PROGRAMME CODE: PGCSS)

Degree Programme for the students admitted from the Academic year 2019-2020 Onwards

PG DEPARTMENT OF COMPUTER SCIENCE

ARULMIGU PALANIANDAVAR ARTS COLLEGE FOR WOMEN

INSTITUTIONAL VISION AND MISSION

VISION

- Enlightenment and Empowerment of Rural Women

MISSION

- To imbibe research activity and collaborative programs with our local communities.
- High quality teaching, providing learning environment with practical exposure.
- Encouragement of a questioning spirit and self-reliance.
- Strong and support education for the students employability

PG DEPARTMENT OF COMPUTER SCIENCE

VISION

- Employing women in the field of Information Technology.
- Molding rural women into Future Leaders.

MISSION

- Training students in latest trends in IT Field.
- Motivating students to organize IT related competitions.
- Conducting special lectures for the students to advance the state of the art in computer science and IT Field.
- Training students to do projects in recent technologies.

M.Sc (Computer Science)

REGULATION FOR ADMISSION

1. **Preamble**

Computer Science department was established in 1988 as self supporting department with the curriculum specifically designed to reflect the depth and breadth of computer science. To encourage young rural women students to adopt higher education, M.Sc Computer Science Programme is added in the department in 2005. Expert members from Academia and Industry provide inputs in introducing specialized courses in the curriculum to suit industry needs. To further enhance the quality of the programme, the department adheres Outcome Based Education (OBE) 2019-2020 onwards.

2. **Eligibility for Admission**

Candidates for admission to the M.Sc (Computer Science) course (Full-Time) should possess a B.Sc (Computer Science) / B.Sc(IT)/ B.Sc(CT) / BCA or any equivalent degree with a minimum aggregate of 55% marks in Part III Subjects.

3. **Duration of the Course**

Full-Time M.Sc (Computer Science) Degree course shall be divided into four semesters of two years duration.

4. **Eligibility for the Degree**

- Candidates for the degree shall besides undergoing the prescribed course of the study, do practical work like case study, project report, prescribed field training etc., under the guidance of staff members and the Head of the Department.
- No candidate shall be eligible for the degree unless she has completed the prescribed course of the study in an Institution and has passed the prescribed examinations.
- No candidate shall be admitted to the examination unless she has put in not less than 60% attendance in terms of total number of working days and has produced a certificate from the Head of the Institution where she has studied that her progress and conduct have been satisfactory.

5. **Passing Rules**

75% of marks are allotted for external evaluation and 25% of the marks are allotted for internal evaluation in each of the theory subjects.

60% of marks are allotted for external evaluation and 40% of the marks are allotted for internal evaluation in each of the practical subjects.

A Candidate is deemed to have passed in a subject if she gets a **minimum of 50%** of the total marks taking the University examination.

6. Distribution of Marks for External Examinations

Course	External (75 Marks)	Average of Passing Minimum
PG	38/75(50%)	50/100

7. Pattern of Evaluation

For each paper there will be Internal Assessment (IA) and Semester Examination (External).

	Int.	Ext.	Total
Theory	25	75	100
Practical	40	60	100
Project	50	150	200

8. Internal Assessment Components

Theory

Test	-	15
Assignment	-	5
Seminar	-	5

		25
		---

9. Content Delivery methods

- Lecture method
- Group Discussion
- ICT

10.Pattern of the Question Paper(Internal)

Maximum: 30 marks Time: 2 Hours

Part – A

I. Answer the following questions(Either or Choice) (2*5=10)

Part – B

II. Answer the following questions(Either or Choice) (2*10=20)

11.Pattern of the Question Paper(External)

Maximum: 75 marks Time: 3 Hours

Part – A

I. Answer any FIVE out of EIGHT questions (5*3=15)

Each unit must have ONE or TWO questions

Part – B

II. Answer the following questions (Either or Choice) (5*6=30)

ONE question from each unit

Part – C

III. Answer any THREE out of FIVE questions (3*10=30)

ONE question from each unit

BLOOM'S TAXONOMY BASED ASSESSMENT PATTERN

K1-Knowledge; **K2**- Comprehension; **K3**- Application; **K4**-Analysis;
K5- Synthesis; **K6**-Evaluation;

1. Theory – 75 Marks

Cognitive Level	Section	Marks	Course Outcomes	Description	Total
K2, K3	A (5 Out of 8)	5 x 3 = 15	CO1,CO2	Short Answers	75
K4	B (Either or Choice)	5 x 6 = 30	CO2,CO3	Descriptive / Detailed	
K5,K6	C (3 Out of 5)	3 x 10 = 30	CO4,CO5	Descriptive / Detailed	

2. Practical Examinations:

Knowledge Level	Section	Marks
K3	Practical & Record Work	60
K4		
K5		

Bloom's Taxonomy in fixing the Learning Objectives:

Since the Academic year 2019 – 2020, the curriculum of B.Sc., (Computer Science) and M.Sc., (Computer Science) has been designed and the learning objectives and outcomes of the programmes are set, following the Bloom's Taxonomy Cognitive Domain. Accordingly, it is broken into six levels of learning objectives of each course. They are -

K1 / Knowledge = Remember

K2 / Comprehension = Understand

K3 / Application = Apply

K4 / Analysis = Analyze

K5 / Evaluation = Evaluate

K6 / Synthesis = Create

Bloom's Taxonomy Action Verbs:

K1 / Knowledge: Arrange, Define, Describe, Duplicate, Identify, Label, List, Match, Memorize, Name, Order, Outline, Recognize, Relate, Recall, Repeat, Reproduce, Select, State

K2 / Comprehension: Classify, Convert, Defend, Describe, Discuss, Distinguish, Estimate, Explain, Express, Extend, Generalize, Give example(s), Identify, Indicate, Infer, Locate, Paraphrase, Predict, Recognize, Rewrite, Review, Select, Summarize, Translate

K3 / Application: Apply, Change, Choose, Compute, Demonstrate, Discover, Dramatize, Employ, Illustrate, Interpret, Manipulate, Modify, Operate, Practice, Predict, Prepare, Produce, Relate, Schedule, Show, Sketch, Solve, Use, Write

K4 / Analysis: Analyze, Appraise, Breakdown, Calculate, Categorize, Compare, Contrast,

Criticize, Diagram, Differentiate, Discriminate, Distinguish, Examine, Experiment, Identify, Illustrate, Infer, Model, Outline, Point out, Question, Relate, Select, Separate, Subdivide, Test

K5 / Evaluation: Appraise, Argue, Assess, Attach, Choose, Compare, Conclude, Contrast, Defend, Describe, Discriminate, Estimate, Evaluate, Explain, Judge, Justify, Interpret, Relate, Predict, Rate, Select, Summarize, Support, Value

K6 / Synthesis: Arrange, Assemble, Categorize, Collect, Combine, Comply, Compose, Construct, Create, Design, Develop, Devise, Explain, Formulate, Generate, Plan, Prepare, Rearrange, Reconstruct, Relate, Reorganize, Revise, Rewrite, Set up, Summarize, Synthesize, Tell, Write

Mapping COs with POs:

For each programme, the Educational objectives and the Specific objectives are specified. The programme outcomes are designed according to the curriculum, teaching, learning and evaluation process. For each course, the definite outcomes are set, giving challenge to the cognitive domain. The course outcomes are mapped with the programme outcomes. The performance of the stakeholders is assessed and the attainment rate is fixed, by using the measurements ‘high’, ‘medium’ and ‘low’. The restructuring of the curriculum is done based on the rate of attainment.

Institutional Objectives:

- Women Education
- Women Empowerment
- Self-reliance and
- Making Model Citizens.

PROGRAM EDUCATIONAL OBJECTIVES

After few years from the completion of M.Sc programme, the students will be able to

PEO1 Work productively as successful Computer professionals in diverse career paths including supportive and leadership roles on multidisciplinary teams or be active in higher studies

PEO2 Communicate effectively, recognize and incorporate societal needs and constraints in their professional endeavors and practice their profession with high regard to Ethical responsibilities

PEO3 Engage in life-long learning and to remain current in their profession to foster personal and organizational growth.

PROGRAM SPECIFIC OUTCOMES

Graduate with a M.Sc. in Computer Science will have the ability to

PSO1 Apply standard Computer science practices and strategies in real-time software project development using open-source programming environment or commercial environment to deliver quality product for the organization success

PSO2 Design and develop computer programs/computer-based systems in the areas related to algorithms, networking, web design, Grid and cloud computing

PSO3 Able to pursue research in Data mining, Image processing and Networking areas and implement his work in MATAB and **.**Net environment .

Mapping PEOs with IOs:

Programme Educational Objectives	Institutional Objectives			
B.Sc., (Computer Science)	**1**	**2**	**3**	**4**
PEO1:.	*			
PEO2:		*		
PEO3:.			*	
PEO4:				*

Measuring: H – High; M – Medium; L – Low

M.Sc., (COMPUTER SCIENCE)

PROGRAM OUTCOMES

Upon successful completion of the programme, the student will able to

PO1 Communicate computer science concepts, designs, and solutions effectively and professionally.

PO2 Apply knowledge of computing to produce effective designs and solutions for specific problems.

PO3 Identify, analyze, and synthesize scholarly literature relating to the field of computer science.

PO4 Use software development tools, software systems, and modern computing platforms.

PO5 Attend SET/NET exams with confidence

SEMESTER-WISE DISTRIBUTION OF COURSES WITH SCHEME OF VALUATION

UNDER CBCS PATTERN

OUTCOME BASED EDUCATION (OBE)

SEMESTER	SUBJECT CODE	TITLE OF THE PAPER	HOURS	CREDITS	MARKS			Exam (Hrs)
					INT.	EXT.	TOT.	
I	**MPCSC1**	**Core Paper I:**Mathematical Foundation of Computer Science	5	4	25	75	100	3
	MPCSC2	**Core Paper II:** Advanced Java Programming	5	4	25	75	100	3
	MPCSC3	**Core Paper III:** Advanced Operating System	5	4	25	75	100	3
	MPCSL1	**Core IV - Practical I:** Advanced Java Programming Lab	5	4	40	60	100	3
	MPCSL2	**Core V - Practical II:** Web Technology Lab	5	4	40	60	100	3
	MPCSE1	**Elective I**	5	4	25	75	100	3
		Total	**30**	**24**			**600**	
II	**MPCSC4**	**Core Paper VI:** Cryptography & Network Security	5	4	25	75	100	3
	MPCSC5	**Core Paper VII:** Data Mining	5	4	25	75	100	3
	MPCSC6	**Core Paper VIII:** Python Programming	5	4	25	75	100	3
	MPCSL3	**Core IX - Practical III:** Python Programming Lab	5	4	40	60	100	3
	MPCSL4	**Core X - Practical IV:** .NET Programming Lab	5	4	40	60	100	3
	MPCSE2	**Elective II**	5	4	25	75	100	3
		Total	**30**	**24**			**600**	
		Internship Training ***(15 Days)***						

III	**MPCSC7**	**Core Paper XI:** Big Data Analytics	5	4	25	75	100	3
	MPCSC8	**Core Paper XII:** Software Project Management	5	4	25	75	100	3
	MPCSC9	**Core Paper XIII:** PHP and MYSQL	5	4	25	75	100	3
	MPCSL5	**Core XIV- Practical V:** PHP and MYSQL Lab	5	4	40	60	100	3
	MPCSL6	**Core XV - Practical VI:** MATLAB	5	4	40	60	100	3
	MPCSE3	**Elective III**	5	4	25	75	100	3
		Internship Training	**-**	GRADE	**-**	**-**	100	**-**
		Total	**30**	**24**			**700**	
IV	**MPCSC10**	**Core Paper XVI:** Digital Image Processing	5	4	25	75	100	3
	MPCSE4	**Elective IV**	5	4	25	75	100	3
	MPCSPR	**Core XVII:** Project Work	20	10	50	150	200	3
		Total	**30**	**18**			**400**	
		Grand Total	**120**	**90**			**2300**	

DISTRIBUTION OF CORE AND ELECTIVE

CORE : 17 (10 Theory + 6 Lab + 1 Project)

ELECTIVE : 04

INTERNSHIP TRAINING : 1

TOTAL MARKS : 2300

TOTAL CREDITS : 90

SEMESTER I

Programme : M.Sc.,

Subject : Computer Science

Semester: I

Course : Mathematical Foundation of Computer Science

Course Type : Core I

Credits : 4

Contact Hours: 5 hours/week

Course Code : MPCSC1

CIA : 25

CE : 75

Preamble

The main objective of this course is to introduce the basic terminology used in Computer application. This emphasizes the development of rigorous logical thinking for solving different kinds of problems that occur in computer applications. This course aims at giving adequate exposure in the theory and applications of Set theory, relations, functions, graph theory and Automata theory.

Course Outcomes

On successful completion of the course, the students will be able to

- Recognize mathematical logics to solve computational problems
- Examine the concepts of sets, relations and functions
- Formulate problems and solve recurrence relations
- Develop solutions for real world problems using graph theory

COURSE CONTENT

UNIT-I :

Mathematical Logic: Introduction – Propositions – Connectives – Order of Precedence for Logic Connectives – Conditional and Bi-Conditional Propositions – Tautology and Contradiction – Equivalence of Propositions – Duality Law – Algebra of Propositions – Tautological Implication – Normal Forms.

UNIT-II :

Set Theory: Introduction – Basic Concepts and Notations – Ordered Pairs and Cartesian Products – Set Operations.

Relations: Type of Relations – Some Operations on Relations – Composition of Relations – Properties of Relations – Partition of a Set - Matrix Representation of Relations by Graphs – Representation of Relations by Graphs.

UNIT-III :

Functions: Introduction – Representation of a Function – Types of Functions – Classification of Functions –Composition of Functions – Inverse of Function – Binary n-ary Operations –Properties of Binary Operations.

UNIT-IV :

Graph Theory: Introduction –Basic Definitions –Degree of a Vertex – Some Special Simple Graphs - Matrix Representation of Graphs – Paths, Cycles and Connectivity – Eulerian and Hamiltonian Graphs – Connectedness in Directed Graphs – Shortest Path Algorithms.

UNIT-V :

Formal Languages and Automata Theory: Introduction –Phrase – Structure Grammar – Types of Phrase –structure Grammar – Backus-Naur Form (BNF) .

Finite State Machine: Input and Output Strings for FSM – Finite State Automata(FSA)- Push down Automaton – Turing Machine.

PRESCRIBED TEXT

- T.Veerarajan, *Discrete Mathematics with Graph Theory and Combinatorics*, McGraw Hill, 2007.

REFERENCE BOOK

- C.L.Liu, *Elements of Discrete Mathematics*, McGraw Hill, 1985.

****

Programme : M.Sc.,

Subject : Computer Science

Semester: I

Course : Advanced java Programming

Course Type : Core II

Credits : 4

Contact Hours: 5 hours/week

Course Code : MPCSC2

CIA : 25

CE : 75

Preamble

This course helps the students to understand the advanced features of java programming such as applet, servlet, java beans, RMI and JDBC.

Course Outcomes

On successful completion of the course, the students will be able to

- Understand the logics of applets, AWT event handling, Servlet and RMI
- Write servlets to access database using Java Data Base Connectivity (JDBC)
- Demonstrate capabilities of server using the concept of Servlet
- Validate remote methods in an application using Remote Method Invocation (RMI)

COURSE CONTENT

UNIT-I :

Applet And Graphics: Applet life cycle - Applet methods - Passing parameters to Applets - getDocumentBase() and getCodeBase() - Using images - Applet interfaces - Difference between Applet and Application Program - Drawing lines and different Shapes - Clipping.

UNIT-II :

AWT And Event Handling: Introduction – Component – Frame - Button class - Layout Management – Insets – Canvas – Label - Text field- Check box - Check box group – Choice – List – Menu - Event handling - Adapter class.

UNIT-III :

Servlet: Introduction – DHTML - CGI Script - Java Servlet - Servlet Container - Servlet Life Cycle - Servlet Interface - Generic Servlet class - HttpServlet Class - HttpServlet Interface - getOutputStream method - setHeader() method - parameter passing to servlet - More about Servlet Owner - Java Web Server and Cookies.

UNIT-IV :

JDBC: Introduction – Introduction to SQL – Database connectivity – ODBC and JDBC API – JDBC application architecture – Steps for creating ODBC DSN – The statement object – Working with Resultset

Java Beans: Introduction – Getting started for Beans – Using BDK Demonstration Beans – Saving and restoring Beans – Building an applet from BeanBox – Create your own bean – InfoBus – Java Activation Framework – Enterprise JavaBeans – The JavaBean bridge for ActiveX – Other Bean Development Tools

UNIT-V :

Networking: TCP/IP - UDP/IP - IP Address – DNS – Port – URL - Socket Programming using TCP/IP and UDP/IP

RMI: Introduction - RMI Packages - Programming using RMI – RMI Enhancements

PRESCRIBED TEXT

- R.Krishnamoorthy and S.Prabhu, *Internet and Java Programming, le*, Publishers: New Age International, 2013.

REFERENCE BOOKS

- Herbert Schildt, *Java2: The Complete Reference,* Tata McGraw Hill, 2013
- Dr. C.Muthu, *Essentials of Java programming*, Vijay Nicole Imprints Pvt. Ltd.,
- K.Somasundaram, *Advanced Java Programming in Java2,* Jaico Publishing House, Mumbai, 2008

****

Programme : M.Sc.,

Subject : Computer Science

Semester: I

Course : Advanced Operating Systems

Course Type : Core III

Credits : 4

Contact Hours: 5 hours/week

Course Code : MPCSC3

CIA : 25

CE : 75

Preamble

This course intended to give students a thorough understanding of design and implementation issues for modern multicore operating systems and helps the students to learn the theoretical foundation of clock, mutual exclusion, deadlock detection, resource sharing and concurrency control in distributed environment.

Course Outcomes

On successful completion of the course, the students will be able to

- Understand the concepts of distributed operating system
- Apply the concepts of synchronization and handle Deadlocks
- Examine the functionalities of distributed resource management
- Evaluate various operating systems such as multiprocessor operating system and database operating system

COURSE CONTENT

UNIT-I :

Process Synchronization: Overview - Functions of an OS – Design approaches. **Synchronization mechanisms:** Concept of a process – concurrent process – Critical section problem – Other synchronization problems. **Process Deadlocks:** Preliminaries – Models of Deadlocks – Models of resources – A Graph theoretic model of a system state – Systems with only reusable Resources.

UNIT-II :

Distributed Operating System: Communication networks – Communication primitives. **Distributed Mutual exclusion:** Preliminaries – Non- Token based and Token Based Algorithms. **Distributed Deadlock detection:** Deadlock handling strategies – Control organization for Distributed Deadlock Detection.

UNIT-III :

Distributed Resource Management: Architecture – Mechanisms – Design Issues – **Distributed shared memory:** Architecture – Algorithms for Implementing DSM – Memory coherence – Coherence protocols – Design Issues. **Distributed scheduling:** Issues in Load Distributing – components – Load distributing algorithms.

UNIT-IV :

Multiprocessor Operating Systems: Motivations – Basic Multiprocessor System Architectures – Interconnection Networks – Caching – MOS Structures – Threads – Process Synchronization – Processor Scheduling – Memory Management.

UNIT-V :

Database Operating Systems: Introduction – Concurrency Control: Introduction -Database Systems : Transactions – Conflicts – Transaction Processing – Serializability Theory – Distributed database systems : Transaction Processing Model - Serializability Condition in DDBS – Data Replication – Complications due to Data Replication – Fully -Replicated Database Systems.

PRESCRIBED TEXT

- Mukesh Singhal, Niranjan G.Shivaratri, *Advanced Concepts in Operating Systems: Distributed, Database and Multiprocessor Operating Systems*, TMH, 2001.

REFERENCE BOOKS

- Andrew S. Tanenbaum, *Modern Operating System*, PHI , 2003.
- Pradeep K.Sinha, *Distributed Operating System concepts and Design*, PHI, 2003

****

Programme : M.Sc.,

Subject : Computer Science

Semester: I

Course : Advanced Java Programming Lab

Course Type : Core IV

Credits : 4

Contact Hours: 5 hours/week

Course Code : MPCSL1

CIA : 40

CE : 60

Preamble

This course helps the students to develop projects using advanced features of java programming such as applet, servlet, java beans, RMI and JDBC.

Course Outcomes

On successful completion of the course, the students will be able to

- Execute interactive web pages using HTML and Javascript
- Acquire knowledge about Servlet and RMI
- To connect java program with external database using JDBC

COURSE CONTENT

PROGRAM LIST

1. Program to display life cycle of an applet.
2. Program to display digital clock using applet
3. Program to display different graphical Shapes in applet.
4. Program to display graphical bar chart by passing parameters in applet
5. Program to find factorial of N using AWT high level event handling.
6. Program to illustrate window closing using AWT low level event handling.
7. Program to find sum of digits using RMI.
8. Program to find length of the given string using RMI.
9. Program to compute factorial of N using Servlet.
10. Program to compute factorial value of N using Servlet.
11. Program to display student mark statement using Servlet and JDBC

****

Programme : M.Sc.,

Subject : Computer Science

Semester: I

Course : Web Technology Lab

Course Type : Core V

Credits : 4

Contact Hours: 5 hours/week

Course Code : MPCSL2

CIA : 40

CE : 60

Preamble

This course enable the students to learn the concepts of designing the websites using HTML, CSS, JavaScript and JSP.

Course Outcomes

On successful completion of the course, the students will be able to

- Create web pages using HTML and CSS
- Apply Javascript for interactive web pages
- Validate server side scripting using JSP

COURSE CONTENT

PROGRAM LIST

1. Simple HTML Page using formatting tags, table, image and frames
2. Design a web page using CSS
3. Prepare a resume using forms
4. Website Design for a Department (or) College (or) Company etc.
5. Create a Link to a CSS to display student information
6. Java Script for a Mathematical Calculator
7. Java Script – Number Puzzle
8. Java script-Games using Random number generation
9. Validation of name, mobile number, date of birth, email id using Java Script
10. Online Quiz using JSP
11. Perform Arithmetic operations using JSP
12. Generate Fibonacci series using JSP

Programme : M.Sc.,

Subject : Computer Science

Semester: I

Course : Design and Analysis of Algorithm

Course Type : Elective 1.1

Credits : 4

Contact Hours: 5 hours/week

Course Code : MPCSE1

CIA : 25

CE : 75

Preamble

This course offers introduction to various common algorithm design techniques, methods for analyzing the performance of corresponding algorithms and improving their efficiency, and to provide performance guarantees.

Course Outcomes

On successful completion of the course, the students will be able to

- Get the idea of working principles of different algorithms
- Understand the concept of various searching and sorting algorithms
- Analyze various design and analysis techniques such as greedy algorithms, dynamic programming, back tracking and branch & bound
- Evaluate time complexity using Asymptotic Notation

COURSE CONTENT

UNIT-I :

Introduction: Algorithm – Algorithm specification - Performance Analysis: Space Complexity -Time complexity– Asympotic Notation (O, Θ, Ω).

Divide and Conquer: General method - Binary search - Merge sort - Quick sort.

UNIT-II :

Greedy Method: The General Method – Knapsack Problem - Job Sequencing with Deadlines. M**inimum-Cost Spanning Trees:** Prim's and Kruskal's algorithms - Single Source Shortest Paths.

UNIT-III :

Dynamic Programming: The General Method - Multistage Graphs -All Pairs Shortest Paths - Single Source Shortest Paths-O/1 Knapsack problem-The Traveling sales-Person Problem.

UNIT-IV :

Basic Traversal and Search Techniques: Techniques for Binary Trees – Techniques for Graphs – Connected Components and Spanning Trees – Biconnected Components and DFS.

UNIT-V :

Back Tracking: The General method – The 8-Queens Problem - Sum of Subsets - Graph Coloring-Hamiltonian Cycles.

Branch-And-Bound: The Method - Least Cost(LC) Search – Bounding - FIFO Branch-and-Bound - LC Branch-and-Bound.

PRESCRIBED TEXT

- Ellis Horrowitz, Sartaj Sahini and Sanguthevar Rajasekaran, *Fundamentals of Computer Algorithms*, 2nd Edition, Universities press.

REFERENCE BOOKS

- Gilles Brassard and Paul Bratley, *Fundamentals of algorithm*, Prentice Hall of India Pvt.Ltd., 1997.
- Mark Allen Weiss, *Data Structures and Algorithms analysis in C*, Addition-Wesley, Third Indian Reprint, 2000.

****

Programme : M.Sc.,
Subject : Computer Science
Semester: I
Course : Embedded System
Course Type : Elective 1.2
Credits : 4
Contact Hours: 5 hours/week
Course Code : MPCSE1
CIA : 25
CE : 75

Preamble

This course helps the students to acquire the knowledge about principles and programming concepts of Embedded system.

Course Outcomes

On successful completion of the course, the students will be able to

- Understand the basic concepts of embedded system
- Interpret the working concept of the processor and memory organization of embedded system
- Examine the real time and embedded system operating systems
- Evaluate various programming environment used to develop embedded systems

COURSE CONTENT

UNIT-I :

Introduction to Embedded System: An Embedded System – Processor in the System – Other Hardware Units – Software Embedded into a System – Exemplary Embedded Systems.

UNIT-II :

Processor and Memory Organization: Structural Units in a Processor – Processor Selection for an Embedded System – Memory Selection for an Embedded system – Direct Memory Access – Devices and Buses for Device Networks: I/O Devices – Timer and Counting Devices – Serial Communication and Parallel Communication – Device Drivers and Interrupts Servicing Mechanism: Device Drivers – Device Drivers for Internal Programmable Timing Devices – Interrupt Servicing (Handling) Mechanism – Context, Latency and Deadline.

UNIT-III :

Programming Concepts and Embedded Programming in C and C++: Software Programming in Assembly Language (ALP) and in High Level Language „C‟ – Embedded Programming in C++ - Embedded Programming in Java – Optimization of Memory needs – Inter-Process Communication and Synchronization of Processes, Tasks and Threads: Multiple Processes

in an Application – Problem of Sharing Data by Multiple Tasks and Routines – Inter Process Communication.

UNIT-IV :

Real Time Operating Systems: Real-Time and Embedded System Operating Systems – Interrupt Routines in RTOS Environment: Handling of Interrupt Source Call by the RTOSs - RTOS Task Scheduling Models, Interrupt Latency and Response Time of the Tasks as Performance Metrics – Performance Metric in Scheduling model for Periodic, Sporadic and Aperiodic Tasks – List of Basic Actions in a Preemptive Scheduler and Expected Times taken at a Processor – Fifteen-Point Strategy for Synchronization between the Processors, ISRs, OS Functions and Tasks and for Resource Management – Embedded Linux Internals: Linux Kernel for the Device Drivers and Embedded System – OS Security Issues.

UNIT-V :

Case Study of an Embedded System for a Smart Card – Hardware & Software Co-Design in an Embedded System: Embedded System Project Management – Embedded System Design and Co-Design Issues in System Development Process – Design Cycle in the Development Phase for an Embedded System – Users of Target System or its Emulator and In-Circuit Emulator(ICE) – Use of Software Tools for Development of an Embedded System – Use of Scopes and Logic Analyzers for System Hardware Tests – Issues in Embedded System Design.

PRESCRIBED TEXT

- Raj Kamal, *Embedded Systems – Architecture, Programming and Design*, Tata McGraw-Hill, 2003.

REFERENCE BOOKS

- David E. Simson, *An Embedded Software Primer*, AddisonsWesley-2001.
- Steve Heath, *Embedded Systems Design*, Elsevier, 2003.
- Frank Vahid and Tony Givargis, *Embedded System Design*, John Wiley And Sons, Inc, 2002.

****

Programme : M.Sc.,

Subject : Computer Science

Semester: I

Course : Artificial Intelligence and Machine Learning

Course Type : Elective 1.3

Credits : 4

Contact Hours: 5 hours/week

Course Code : MPCSE1

CIA : 25

CE : 75

Preamble

This course offers knowledge about fundamentals of knowledge representation for problem solving, learning methods of Artificial Intelligence. It also helps the students to understand the deeper concepts of Machine Learning and Algorithms

Course Outcomes

On successful completion of the course, the students will be able to

- Understand the basic concepts of Artificial Intelligence and machine learning algorithms
- Classify strength and weakness of different problem solving techniques
- Apply Artificial Intelligence and Machine Learning Techniques to solve real world problems
- Examine the different heuristic techniques for problem solving and create new solutions

COURSE CONTENT

UNIT-I :

Introduction: Definition of AI - AI Problems – Topics of AI – Production Systems – State space Representation - Applications of AI

UNIT-II :

Heuristic Search Techniques: Generate and Test - Hill Climbing - Search techniques - Problem Reduction – Constraint Satisfaction – Means –end- Analysis

UNIT-III :

Game Playing: MINIMAX Procedure – ALPHA-BETA Pruning – Combined Approach Knowledge representation: – Knowledge Management – types of Knowledge – Knowledge representation – Approaches to knowledge Representation - Issues in Knowledge representation – Reasoning

UNIT-IV :

Learning –Association Learning - clustering: K-means clustering – Fuzzy clustering – Hierarchical Clustering – Reinforcement Learning: Markov Decision Problem - Q- Learning – Learning Automata – Statistical Learning: Hidden Markov Models – Linear Classifiers – Quadratic Classifiers –Decision Trees – Bayesian Networks

UNIT-V :

Supervised Learning: Support Vector - Case-based Reasoning – Ensemble Classifiers – Nearest Neighborhood – Unsupervised Learning: Expectation maximization – Self organizing Maps – Adaptive Resonance Theory

PRESCRIBED TEXT

- S.S. Vinod Chandra, S. Anand Hareendran, *Artificial Intelligence and machine Leaning*, Eastern Economy Edition (2014), PHI Learning Private Limited, New Delhi.

REFERENCE BOOKS

- Elaine Rich and Kevin Knight, *Artificial Intelligence*, 3/e, Tata McGraw Hill (2009), New Delhi.
- Donald A. Waterman, *A Guide to Expert Systems*, Tech knowledge Series in Knowledge Engineering, (2003) New Delhi
- Charnaik, E., C.K. Reiesbeck, and D.V. McDermett, *Artificial Intelligence Programming*, Lawrence Erlbaum Associates (2000), New Jersey
- Stephen Marsland, *Machine Learning: An Algorithmic Perspective*, Chapman and Hall, 2009.

****

SEMESTER II

Programme : M.Sc.,

Subject : Computer Science

Semester: II

Course : Cryptography and Network Security

Course Type : Core VI

Credits : 4

Contact Hours: 5 hours/week

Course Code : MPCSC4

CIA : 25

CE : 75

Preamble

This course provides an essential study of Computer Security Issues and Methods in Networking Systems. It also helps students to identify Ethical, Professional responsibilities, Risks and Liabilities in Computer and Network Environment.

Course Outcomes

On successful completion of the course, the students will be able to

- Describe cryptography and network security concepts and its application
- Get the idea about encryption standards
- Examine various cryptography algorithms
- Validate the authentication using digital signature and authentication protocols

COURSE CONTENT

UNIT-I :

Introduction: The OSI Security Architecture – Security Attacks – Security Services – Security Mechanisms- **Classical Encryption Techniques** : Symmetric Cipher Model – Substitution Techniques – Transposition Techniques

UNIT-II :

Block Ciphers and the Data Encryption Standards : Block Cipher Principles – The Data Encryption Standard– **Advanced Encryption Standards**: Evaluation Criteria For AES – The AES Cipher

UNIT-III :

More on Symmetric Cipher: Multiple Encryption and Triple DES – Block Cipher models of operation. **Confidentially using Symmetric Encryption:** Placement of encryption function – Traffic confidentially – Key distribution – Random number generation

UNIT-IV :

Public Key Cryptography and RSA: Principles of Public Key Crypto systems – The RSA Algorithm – **Key Management** : Key Management – Diffie Hellman Key Exchange

UNIT-V :

Message Authentication and Hash Functions: Authentication Requirements - Authentication Functions –Message Authentication Codes – Hash Functions. **Digital Signatures and Authentication Protocols**: Digital Signatures –Authentication Protocols – Digital Signature Standards.

PRESCRIBED TEXT

- William Stallings, *Cryptography and Network Security Principles and Practices*,4th edition.

REFERENCE BOOK

- William Stallings, *Cryptography and Network Security*, Pearson Publications, 5th edition.

****

Programme : M.Sc.,
Subject : Computer Science
Semester: II
Course : Data Mining
Course Type : Core VII
Credits : 4
Contact Hours: 5 hours/week
Course Code : MPCSC5
CIA : 25
CE : 75

Preamble

This course enables the students to learn the concepts of data mining tasks, classification, clustering and web mining. It also imparts the storage of heterogeneous data in data warehousing.

Course Outcomes

On successful completion of the course, the students will be able to

- Understand the basic Concepts of data mining and data warehousing
- Analyze various data mining techniques like classifications, clustering, association rule mining, prediction and related algorithm
- Choose appropriate data mining techniques to carry out simple data mining tasks
- Develop data mining algorithms to store heterogeneous data

COURSE CONTENT

UNIT-I :

Data Warehousing: Introduction – Data Warehouse Architecture – Dimensional Modelling – Categorisation of Hierarchies – Aggregate Function – Summarisability – Fact-Dimension Relationship – OLAP Operations – OLAP Server – ROLAP – MOLAP.

UNIT-II :

Data Mining: Introduction - Definition – KDD vs Data Mining – DBMS vs DM – DM Techniques – Issues and Challenges in DM – DM Application Areas.

UNIT-III :

Association Rules: Introduction – Methods to Discover Association Rules – A Priori algorithm – Partition Algorithm – Pincer-Search Algorithm – Incremental Algorithm - Border Algorithm – Generalised Association Rules.

UNIT-IV :

Clustering Techniques: Introduction - Clustering Paradigms – Partitioning Algorithms K-Medoid Algorithms – CLARA – CLARANS- Hierarchical Clustering – DBSCAN.

Decision Trees: Introduction - Tree Construction Principle – Best Split – Splitting Indices – Splitting Criteria – CART – ID3.

UNIT-V :

Web Mining: Introduction – Web Content Mining – Web Structure Mining – Web Usage Mining – Text Mining – Hierarchy of Categories – Text Clustering.

PRESCRIBED TEXT

- Arun K Pujari, *Data Mining Techniques,* Universities Press – Fourth Edition**.**

REFERENCE BOOKS

- Jewie Han, Michelins Kamber, *Data Mining: Concepts and Techniques*
- Pang-Ning Tan, Michael Steinbach, Vipin Kumar, *Introduction to Data Mining*, 2007.

****

Programme : M.Sc., **Subject** : Computer Science

Semester: II **Course** : Python Programming

Course type : Core VIII **Credits** : 4

Contact Hours: 5 hours/week **Course Code** : MPCSC6

CIA : 25 **CE** : 75

Preamble

This course helps the students to understand the core concepts of programming in Python such as strings, operators, conditional statements, lists, tuples, dictionaries, functions and files.

Course Outcomes

On successful completion of the course, the students will be able to

- Get the basic knowledge about Python programming
- Apply essential programming concepts like strings, operators, conditional statements, functions, files and exception handling of Python in simple programs
- Analyze various concepts of Python
- Create a applications using core concepts of Python

COURSE CONTENT

UNIT-I :

Introduction to Python: Introduction – Python overview – Getting started with python – Comments – Python identifiers – Reserved keywords – Variables – Standard data types – Operators – Statements and Expressions – String operations – Boolean expressions.

Classes and Objects: Overview of OOP – Class definition – Creating objects –Objects as Arguments – Objects as return value – Build-in class attributes – Inheritance – Method overriding – Data encapsulation – Data hiding.

UNIT-II :

Control Statements: The for loop – While statement – if elif else statement – Input from keyboard.

Functions: Introduction – Built-in functions – Type conversion – Type coercion – Date and time – dir() function – help() function – User defined functions – Parameters & arguments – Function calls – The return statement – Python recursive function – The anonymous functions – Writing python scripts.

UNIT-III :

Strings: Strings – Compound data type – len function – String slices – Strings are Immutable - String traversal – Escape characters – String formatting operator – String formatting functions.

Lists: Values and accessing elements – Lists are Mutable – Traversing a list – Deleting elements from list – Built-in list operators – Built-in list methods.

UNIT-IV :

Tuples: Creating tuples – Accessing values in tuples – Tuple assignment – Tuples as return values – Basic tuple operations – Built-in tuple functions

Dictionaries: Creating dictionary – Accessing values in dictionary – Updating dictionary – Deleting elements from dictionary – Operations in dictionary Built-in dictionary methods.

UNIT-V :

Files and Exceptions: Text Files – Opening a File – Closing a File – File object attributes – Reading from a file – Writing to a file – Renaming a file – Deleting a file – File related methods – Directories – Exceptions – Build-in Exceptions – Handling Exceptions – Exception with arguments – User defined Exceptions.

PRESCRIBED TEXT

- E Balagurusamy, *Introduction to Computing and Problem Solving Using Python*, McGrawHill publication (2018).

REFERENCE BOOKS

- Mark Lutz, *Programming Python*, 4/e, O'Reilly Media (2010).
- Mark Summerfield, *Programming in Python 3*, Pearson Education.

****

Programme : M.Sc.,

Subject : Computer Science

Semester: II

Course : Python Programming Lab

Course Type : Core IX

Credits : 4

Contact Hours: 5 hours/week

Course Code : MPCSL3

CIA : 40

CE : 60

Preamble

This course helps the students to produce well designed programs using concepts of python like strings, lists, tuples, dictionaries, files and exception handling.

Course Outcomes

On successful completion of the course, the students will be able to

- Implement various operators of Python
- Review and analyze the Python programs with variables, loop, functions and operators
- Develop application with Python core concepts

COURSE CONTENT

PROGRAM LIST

1. Exercises using if-else and conditional operators
2. Exercise to implement Recursive and Non-Recursive functions
3. Exercise to implement String
4. Exercises to implement list
5. Exercises using Dictionary.
6. Exercises to perform set operations.
7. Exercises using object oriented concepts.
8. Exercises to implement File handling concept
9. Exercises using exceptional handling technique.
10. Exercise to implement Inheritance and polymorphism

****

Programme : M.Sc.,

Subject : Computer Science

Semester: II

Course : Dot Net Programming Lab

Course Type : Core X

Credits : 4

Contact Hours: 5 hours/week

Course Code : MPCSL3

CIA : 40

CE : 60

Preamble

This course presents the practical aspects of application development using .Net framework. It also concerns about Common Language Runtime (CLR) and SQL Server concepts.

Course Outcomes

On successful completion of the course, the students will be able to

- Develop simple VB.NET program using forms
- Execute VB.NET application with various controls
- Update database using SQL server

COURSE CONTENT

PROGRAM LIST (VB.NET)

1. Write a VB.NET program to receive user feedback using Form and stored it in a database.
2. Write a simple VB.NET program to receive the text and print it using button.
3. Write an VB.NET program to design an application for dynamically populating checkbox list.
4. Create an VB.NET program to design an application using grid view control in a web page.
5. Create an VB.NET applications for
    a. Hospital management
    b. Hotel management,
    c. Online Quiz
    d. Online Shopping
    e. Student Mark Processing System
    f. Online Ticket reservation

****

Programme : M.Sc.,

Subject : Computer Science

Semester: II

Course : Principles of Compiler Design

Course Type : Elective 2.1

Credits : 4

Contact Hours: 5 hours/week

Course Code : MPCSE2

CIA : 25

CE : 75

Preamble

This course offers knowledge about concepts and principles of compiler design. It also provide basic understanding of grammars, language definitions and phases of designing compiler.

Course Outcomes

On successful completion of the course, the students will be able to

- Examine the basic function of compiler and interpreter
- Understand the core concepts of phases of compiler
- Apply Context Free Grammar for simplify the expression using different kinds of parsers
- Interpret the code generation and optimization process

COURSE CONTENT

UNIT-I :

Introduction: The structure of a Compiler – Compilers writing tools- The lexical and Syntactic Structure of a language – The role of the lexical analyzer – The design of lexical analyzers – Regular expressions – Finite automata.

UNIT-II :

Context-free Grammars – Derivation and parse trees – Capabilities of Context Free Grammars – Parses – Shift Reduce parsing - Operators Precedence Parsing – Top-down parsing – Predictive parsers.

UNIT-III :

LR Parsers – The canonical collection of LR (0) Items – Constructing SLR Parsing tables – **Symbol tables:** Data structure for Symbol tables

UNIT-IV :

Syntax-Directed Translation: Syntax-Directed Translation schemes –Implementation of Syntax-directed translators – Intermediate code – Postfix notation – Parse trees and syntax trees– Three-address code, quadruples and triples – Translation of assignment statements – Boolean expressions

UNIT-V :

Introduction To Code Optimization: The Principal source of optimization – Loop optimization – The DAG Representation of Basic Blocks – **Code Generation:** Problems in code generation – A Simple code generator

PRESCRIBED TEXT

- Alfred V. Aho Ravi Sethi Jeffrey D.Ullman, *Principles of Compiler Design*, Published by Narosa Publishing House.

REFERENCE BOOK

- Kennet C.Louden, *Compiler Construction:Principles and Practice*

****

Programme : M.Sc.,

Subject : Computer Science

Semester: II

Course : Client Server Computing

Course Type : Elective 2.2

Credits : 4

Contact Hours: 5 hours/week

Course Code : MPCSE2

CIA : 25

CE : 75

Preamble

This course helps the students to acquire the knowledge about core concept of Client/Server technology.

Course Outcomes

On successful completion of the course, the students will be able to

- Discuss the core concept of Client / Server technology
- Schedule the software required to establish Client/Server computing
- Appraise the Online Transaction processing tools and procedures
- Set up Distributed system management components and design Client/Server Application.

COURSE CONTENT

UNIT-I :

Basic concepts of Client/Server – Characteristics – File Servers – Database servers – Transaction servers- Groupware servers – Objective servers – Web servers – Fat servers or fat clients – 2 tier versus 3 tier – Client/Server building blocks – Operating system services. Base services – Extended services – Server scalability – Client Anatomy.

UNIT-II :

NOS Middleware – Peer-to-peer communications – RPC – MOM Middleware – MOM versus RPC - The fundamentals of SQL and relational databases – Server architecture – Stored procedures, triggers and rules.

UNIT-III :

Online transaction processing – Decision support systems – OLTP versus DSS: programming effort, database needs – Data warehouses – Elements - Hierarchies – Replication versus Direct access – Replication mechanism – EIS/DSS Tools – Client/server transaction processing – transaction models – TP Monitors – Transaction management standards.

UNIT-IV :

Groupware – Components – Distributed objects and components – CORBA: components – Object Management Architecture – Services – Business objects.

UNIT-V :

Client/server Distributed system management – components – Management application – The Internet Management Protocols – OSI Management Framework – The Desktop Management Interface – X/Open Management Standards – Client/Server application development tools – Client/Server Application Design.

PRESCRIBED TEXT

- Robert Orfali, Dan Harkey and Jeri Edwards, *The Essential Client Server Survival Guide*, 2nd edn. Galgotia

REFERENCE BOOKS

- Dawna Travis Dewire, *Client/Server computing*, Tata McGraw Hill.
- Jafferey D. Schank, *Novell''s guide to Client/Server Application and Architecture*, BPB Publications.

****

Programme : M.Sc., **Subject** : Computer Science

Semester: II **Course** : Neural Networks

Course Type : Elective 2.3 **Credits** : 4

Contact Hours: 5 hours/week **Course Code** : MPCSE2

CIA : 25 **CE** : 75

Preamble

This course provides an essential study of Neural Network Techniques. It also helps the students to understand the fundamental concepts of Pattern Recognition.

Course Outcomes

On successful completion of the course, the students will be able to

- Get the idea about basics of Neural network
- Describe the concepts of pattern recognition
- Examine the concept of neuron and multilayer perceptron
- Analyze Kohenen Self–Organizing Networks and Hopfield Networks of Neural network

COURSE CONTENT

UNIT-I :

Introduction: Humans and Computers: The structure of the Brain, Learning in Machines, the Differences.

UNIT-II :

Pattern Recognition: Introduction, Pattern Recognition in Perspective, Pattern recognition – a definition, feature vectors and feature space, discriminate functions, classification techniques. Linear classifiers statistical techniques, Pattern Recognition – a summary.

UNIT-III :

The Basic Neuron: Introduction - Modeling the single neuron, learning in simple neurons, the perception a vectorial perspective, the perception learning rule, proof, and limitations of perception.

The Multilayer Perceptron: Introduction, altering the perception model, the new model the learning rule, the multiplayer perception algorithm, the XOR problem revisited applications of multiplayer perception.

UNIT-IV :

Kohenen Self – Organizing Networks: Introduction, the kohenen algorithm, weight training neighborhoods, reducing the neighborhoods, learning vector quantization, the Phonetic typewriter.

Hopfield Networks: The Hopfield model, the energy landscape, the Boltzsman machine, constraint satisfaction.

UNIT-V :

Adaptive Resonance Theory: Adaptive resonance theory, architecture and operation, ART algorithm, training the ART network, clarification, conclusion, summary of ART. Hardware and Software implementations, Optical Computing, Optical Computing and neural networks.

PRESCRIBED TEXT

- Adam Hilger, *Neural Computing: An introduction R.Beale&T.Jackson*, 1900.
- James A.Freeman, David, M.Skapura, *Neural Networks-Algorithm, Application and Programming techniques*, Pearson Education.

REFERENCE BOOK

- Fredic M.Ham, IvicalKostanic, *Principles of Neuro computing for science of engineering*, TMCH.

****

SEMESTER III

Programme : M.Sc.,
Subject : Computer Science
Semester: III
Course : Big Data Analytics
Course Type : Core XI
Credits : 4
Contact Hours: 5 hours/week
Course Code : MPCSC7
CIA : 25
CE : 75

Preamble

This course offers knowledge about concepts and challenge of big data. It also the helps students to acquire knowledge about Hadoop, MapReduce, Pig and Hive technology.

Course Outcomes

On successful completion of the course, the students will be able to

- Understand the concept of Big data techniques, environment, framework and Hadoop ecosystem
- Apply Statistical data analysis and tools to manage and analyze the big data
- Analyze Hadoop components and their uses for big data processing
- Examine the impact of big data for business decisions and strategy
- Manage large-scale analytics tools to solve some open big data problems

COURSE CONTENT

UNIT-I :

INTRODUCTION TO BIG DATA: Introduction to Big Data Platform – Challenges of Conventional Systems - Intelligent data analysis Nature of Data - Analytic Processes and Tools - Analysis vs Reporting - Modern Data Analytic Tools

UNIT-II :

STATISTICAL DATA ANALYSIS: Parameter and Statistic- Sampling Distribution-Meaning-Standard Error and its uses - Tests of Significance- Null and Alternative Hypotheses - Type-I and Type-II Error- Critical Region and Level of Significance - One tailed and Two tailed Tests. Critical values or Significant values - Tests of Significance for Large Samples - Test of Significance for Single Proportion - Test of Significance for Difference of Proportions-Test of Significance for Single Mean- Test of Significance for Difference of Means - Chi-Square Distribution – Definition- Applications of Chi-Square Distribution -To test the goodness of fit.- To test the independence of Attributes - Student's" t" - Distribution-Definition- Applications of

Student's "t" –Distribution- To test for Single Mean- To test for Difference of Means- Paired t-test for Difference of MeansF-Distribution- Definition- To Test for Equality of Two Population variances - Meaning of Resampling and its uses. Basic ideas of Randomization, Exact Test, Cross ValidationJacknife and Bootstrap. Prediction Error and its uses.

UNIT-III :

HADOOP: History of Hadoop- The Hadoop Distributed File System – Components of Hadoop-Analyzing the Data with Hadoop- Scaling Out- Hadoop Streaming- Design of HDFS-Java interfaces to HDFS- How MapReduce Works-Anatomy of a MapReduce Job run-Failures-Job Scheduling-Shuffle and Sort – Task execution – MapReduce Features

UNIT-IV :

HADOOP ENVIRONMENT: Setting up a Hadoop Cluster - Cluster specification - Cluster Setup and Installation - Hadoop Configuration-Security in Hadoop - Administering Hadoop – HDFS - Monitoring-Maintenance

UNIT-V :

FRAMEWORKS: Applications on Big Data Using Pig and Hive – Data processing operators in Pig – Hive services – HiveQL – Querying Data in Hive - fundamentals of HBase and ZooKeeper.

PRESCRIBED TEXT

- Michael Berthold, David J. Hand, *Intelligent Data Analysis*, Springer. (2007) (For Unit I to III)
- Tom White (2012), *Hadoop:The Definitive Guide*, Third Edition, O'reilly Media (For Unit IV to V)

REFERENCE BOOKS

- AnandRajaraman and Jeffrey David Ullman, *Mining of Massive Datasets*, Cambridge University Press. (2012)
- Viktor Mayer, Schonberger, Kenneth Cukier, *Big Data : A Revolution That Will Transform How We Live, Work and Think*, Houghton Mifflin Harcourt publishing company.

****

Programme : M.Sc., **Subject** : Computer Science

Semester: III **Course** : Software Project Management

Course Type : Core XII **Credits** : 4

Contact Hours: 5 hours/week **Course Code** : MPCSC8

CIA : 25 **CE** : 75

Preamble

This course provides an essential need for software project management and also highlight the different techniques for software cost estimation and activity planning.

Course Outcomes

On successful completion of the course, the students will be able to

- Review the basic concepts of software project management
- Get the idea about various SDLC models
- Apply forward pass and backward pass to find critical path
- Analyze the behaviours of team members in an organization
- Acquire knowledge about cost monitoring and contract management

COURSE CONTENT

UNIT-I :

Introduction to Software Project Management: Project Definition – Contract Management and Technical Project Management – Activities Covered By Software Project Management – Overview of Project Planning – Stepwise Project Planning.

UNIT-II :

Selection of Appropriate Project Approach: Introduction – Build or Buy? – Choosing Methodologies and Technologies – Software Processes and Process Models – Choice of Process Models – Structure versus Speed of Delivery – The Waterfall Model – The Spiral Model – Software Prototyping – Other ways of Categorizing Prototypes – Incremental Delivery – Rapid Application Development – Agile Methods.

UNIT-III :

Activity Planning: Objectives – Project Schedule – Sequencing And Scheduling Activities – Network Planning Models – Forward Pass – Backward Pass – Activity Float – Shortening Project Duration – Activity on Arrow Networks – Risk Management Categories Of Risk - Risk Management

UNIT-IV :

Monitoring And Control: Creating Framework – Collecting The Data – Visualizing Progress – Cost Monitoring – Earned Value – Prioritizing Monitoring – Getting Project Back To Target – Change Control – Managing Contracts – Introduction – Types Of Contract – Stages In Contract Placement – Typical Terms Of A Contract – Contract Management – Acceptance.

UNIT-V :

Managing People And Organizing Teams : Introduction – Understanding Behavior – Organizational Behaviour: A Background – Selecting The Right Person For The Job – Instruction In The Best Methods – Motivation – The Oldman – Hackman Job Characteristics Model – Working In Groups – Becoming A Team – Decision Making – Leadership – Organizational Structures – Stress – Health And Safety

PRESCRIBED TEXT

- Bob Hughes, *Software Project Management,* Mike Cotterell, Rajib Mall- Fifth Edition McGraw Hill - 2002.

REFERENCE BOOKS

- Andrew Stellman and Jeniffer Greene, *Applied Software Project Management,* O'Reilly Media Inc., - 2006.
- Robert T. Futrell, Donald F. Shafer, Linda Shafer, *Quality Software Project Management,* Prentice Hall – 2002.
- Pankaj Jalote, *Software project management in practice,* Pearson Education - 2004.
- Joel Henry, *Software project management A real-world guide to success,* Pearson/Addison Wesley – 2004.

****

Programme : M.Sc.,
Subject : Computer Science
Semester: III
Course : PHP and MYSQL
Course Type : Core XIII
Credits : 4
Contact Hours: 5 hours/week
Course Code : MPCSC9
CIA : 25
CE : 75

Preamble

This course introduces the concepts of PHP, HTML and MYSQL. It provides concepts of sessions and cookies to develop web pages and the basics of data manipulation using MySQL database.

Course Outcomes

On successful completion of the course, the students will be able to

- Understand the basic concepts of PHP and MySQL
- Illustrate String, array, mathematical, date and time functions in PHP
- Examine the regular expressions and file system of PHP
- To validate database queries in PHP using MySQL

COURSE CONTENT

UNIT-I :

Introduction: Server - Side Web Scripting - Syntax and Variables - Control and Functions. Passing Information between Pages: GET Arguments - POST Arguments - Formatting Form Variables - PHP Super global Arrays.

UNIT-II :

String: Strings in PHP - String Functions-Arrays and Array Functions: Creating Arrays - Retrieving Values - Multidimensional Arrays - Inspecting Arrays - Deleting from Arrays - Iteration. Advanced Array Functions: Transformation of Arrays. Number Handling: Numerical Types – Mathematical Operators - Simple Mathematical Functions - Randomness

UNIT-III :

Regular Expressions: Tokenizing and parsing Functions - Regular Expressions - Perl – Compatible Regular Expressions - Advanced String Functions. Working with the File system: PHP File Permissions - File Reading and Writing Functions - File system and Directory Functions – Network Functions - Date and time Functions - Calendar Conversion Functions. Working with Sessions and Cookies: Sessions work in PHP - Session Functions - Configuration Issues - Cookies – Sending HTTP Headers

UNIT-IV :

Structured Query Language (SQL): Relational Database and SQL - SQL standards – The Workhorses of SQL - Database Design - Privileges and Security. PHP and MYSQL: Connecting to MySQL - Making MySQL Queries - Fetching Data Sets - Multiple Connections - Error Checking - Creating MySQL Databases with PHP - MySQL Functions

UNIT-V :

Performing Database Queries: HTML Tables and Database Tables - Complex mapping – Creating the sample Tables. Integrating Web Forms and Databases: HTML Forms - Basic Form Submission to a Database - Self Submission - Editing Data with an HTML Form

PRESCRIBED TEXT

- Steve Suehring, Tim Converse and Joyce Park, *PHP6 and MySQL Bible*, Wiley-India, New Delhi. (2012)

REFERENCE BOOKS

- Mike McGrath, *PHP and MySQL*, McGraw Hill Education Private Limited, India. (2012)
- Beighley, *Head First Php& MySQL*, O'Reilly Publisher. (2011)
- W. Jason Gilmore, *Beginning PHP and MYSQL: From Novice to Professional*, Dreamtech Press. (2010)

****

Programme : M.Sc.,

Subject : Computer Science

Semester: III

Course : PHP and MYSQL Lab

Course Type : Core - XIV

Credits : 4

Contact Hours: 5 hours/week

Course Code : MPCSL5

CIA : 40

CE : 60

Preamble

This course helps the students to produce well designed programs using PHP and access database using MYSQL

Course Outcomes

On successful completion of the course, the students will be able to

- Apply String functions and regular expressions
- Demonstrate the database connectivity with MYSQL database
- Develop web pages using PHP and MYSQL

COURSE CONTENT

PROGRAM LIST

1. Exercise to pass information between web pages using GET and POST methods.
2. Exercise using arithmetic operations, String functions.
3. Exercise to apply advanced string functions to manipulate strings.
4. Exercise to implement file concepts to open, read, close and to delete a file.
5. Exercise using Regular expressions for validation.
6. Exercise to implement the date and time functions.
7. Exercise to manipulate data using different queries.
8. Exercise to implement explode and implode functions.
9. Create data base connectivity between PHP and MYSQL.
10. Create web pages with PHP and MYSQL database.

****

Programme : M.Sc., **Subject** : Computer Science

Semester: III **Course** : Mat Lab

Course Type : Core XV **Credits** : 4

Contact Hours: 5 hours/week **Course Code** : MPCSL6

CIA : 40 **CE** : 60

Preamble

This course provides the practical solution for mathematical problems and do image enhancement process.

Course Outcomes

On successful completion of the course, the students will be able to

- Illustrate simple mathematical functions/equations in MATLAB
- Interpret and visualize simple mathematical functions and operations thereon using plots or display
- Test the overall structure of MATLAB program to display required output

COURSE CONTENT

PROGRAM LIST

1. Plotting
2. Adding Two Images
3. Crop Image
4. Comparison of Two Values
5. Gray Scale to Binary Image
6. Flipping An Image
7. Image Complement
8. Adding and Reducing Noise in an Image
9. Color Space Conversion
10. Image Reduction
11. Multiply and Divide an Image
12. Image Transformation by DCT
13. Shearing an Image
14. Enhance Contrast in an Image

15. Edge Detection in an Image

****

Programme : M.Sc., **Subject** : Computer Science

Semester: III **Course** : Mobile Computing

Course Type : Elective 3.1 **Credits** : 4

Contact Hours: 5 hours/week **Course Code** : MPCSE3

CIA : 25 **CE** : 75

Preamble

This course introduces the fundamentals of mobile computing and its architecture. It also discusses topics such as mobile communications, wireless application protocol and wireless LAN.

Course Outcomes

On successful completion of the course, the students will be able to

- Understand the concept of communication medium and multiplexing in telephone network
- Comprehend the routing mechanism and frequency allocation in GSM
- Deploy the GPRS concept for packet data transfer in mobile by using GPRS
- Acquire the knowledge on WAP, CDMA, 3G network and spectrum techniques in wireless network

COURSE CONTENT

UNIT-I :

Introduction: Mobility of Bits and Bytes-Wireless The beginning-Mobile computing-Dialog Control-Networks-Middleware and Gateways-Applications and Services-Developing Mobile Computing Applications –Security in Mobile Computing –Standards-why is it necessary-Standard Bodies-Players in the wireless space. **Mobile computing Architecture**: History of computers-History of Internet-Internet the Ubiquitous Network - Architecture for Mobile computing - Three-tier Architecture - Design considerations for mobile computing - mobile computing through Internet - Making existing applications mobile-enabled.

UNIT-II :

Mobile Computing Through Telephony: Evolution of Telephony - Multiple Access Procedures – mobile computing through telephone – developing an IVR Application - voice XML - Telephony Application Programming Interface. **Emerging Technologies:** Introduction -Bluetooth - radio Frequency Identification – wireless broadband - mobile IP - Internet Protocol version 6 - Java card.

UNIT-III :

Global System For Mobile Communications: Global system for Mobile communications – GSM Architecture – GSM Entities – call routing in GSM – PLMN Interfaces – GSM address and Identifiers – Network aspects in GSM - GSM Frequency Allocation – Authentication and security. **General Packet Radio Service**: Introduction – GPRS and packet Data Network - GPRS Network operations – Data Services in GPRS – Applications for GPRS -Limitations of GPRS - Billing and charging in GPRS

UNIT-IV :

Wireless Application Protocol: Introduction – WAP – MMS - GPRS applications. **CDMA and 3G**: Introduction - Spread spectrum technology – IS95 – CDMA versus GSM – Wireless Data – Third Generation Networks – Applications on 3G

UNIT-V :

Wireless LAN: Introduction – wireless LAN advantages – IEEE 802.11 standards – wireless LAN architecture – mobility in wireless LAN **–** deploying wireless LAN – Mobile adhoc Networks and sensor Networks – Wireless LAN Security – WiFi versus 3G.

PRESCRIBED TEXT

- Asoke K Talukder, Roopa R Yavagal, *Mobile Computing, Technology Applications and Service creation,* Tata McGraw - Hill Publishing company New Delhi 2007.

REFERENCE BOOK

- Jochen Schiller, *Mobile Communication,* 2nd edition Pearson 2003.

****

Programme : M.Sc.,
Subject : Computer Science
Semester: III
Course : Software Engineering
Course Type : Elective 3.2
Credits : 4
Contact Hours: 5 hours/week
Course Code : MPCSE3
CIA : 25
CE : 75

Preamble

This course imparts the skills of developing software for various applications and systems. It discusses the various software development life cycle models and testing methodologies.

Course Outcomes

On successful completion of the course, the students will be able to

- Discuss the stages of software development life cycle
- Examine various Software development life cycle models
- Analyze the role of project management including requirement gathering, planning, designing and maintenance
- Evaluate various testing principles on software project for risk management

COURSE CONTENT

UNIT-I :

The Evolving Role of Software – Definition of Software Engineering – The Changing Nature of Software – Software Myths – Terminologies – Software Life Cycle Models: Build and Fix Model – Evolutionary Process Models – Selection of a Life Cycle Model.

UNIT-II :

Requirements: Analysis and Specifications: Type of Requirements – Feasibility Studies – Requirement Elicitation: interviews, brain storming sessions, FAST – Requirement analysis: Data flow diagram, Data Dictionaries - Requirements Validation

UNIT-III :

Project Planning: Size Estimation – The Constructive Cost Model (COCOMO) – The Putnam Resource Allocation Model.

UNIT-IV :

Software Design: Design: Conceptual and Technical designs, Objectives of design – Modularity - Function Oriented Design – Software reliability: Basic concepts, software reliability, maturity levels - Software Testing: A Strategic Approach to Software Testing – Testing – Functional Testing – Structural Testing – Levels of Testing – Validation Testing.

UNIT-V :

Software Maintenance: Categories of Maintenance – Problems during Maintenance – Maintenance is Manageable – Potential Solutions to maintenance problems – Maintenance process – Estimation of maintenance cost

PRESCRIBED TEXT

- K.K.Agarwal, *Software Engineering*, Third Edition 2008

REFERENCE BOOK

- Richard e.Fairley, *Software Engineering Concepts*, McGrawHill

****

Programme : M.Sc.,
Subject : Computer Science
Semester: III
Course : Robotics
Course Type : Elective 3.3
Credits : 4
Contact Hours: 5 hours/week
Course Code : MPCSE3
CIA : 25
CE : 75

Preamble

This course introduces the basic concepts of robotic technology and its classification. It also helps the students to develop small automatic applications with the help of Robotics.

Course Outcomes

On successful completion of the course, the students will be able to

- Get the idea about automation and robots
- Interpret the basic concepts of robotic technology and its classification
- Analyze various types of grippers
- Collect knowledge about robot sensor and vision
- Evaluate robot control systems and safety

COURSE CONTENT

UNIT-I :

Introduction: Objectives - automation and robots - brief history - the technology of robots – economic and social issues - present and future applications.

Robot Technology: Objectives-fundamentals general characteristics - basic components - robot anatomy-robot generations-robot selection.

Robot classification: objectives - classification-arm geometry - degrees of freedom-power sources - type of motion - path control - intelligence level

UNIT-II :

Robot system analysis: Objectives-robot operation - hierarchical control structures - control structure - line tracking - dynamic properties of robots - modular robot components.

Robot End Effectors: objectives - types of end effectors-mechanical grippers - gripper forces analysis - other types of grippers-special-purpose grippers-grippers selection and design-process tooling-compliance.

UNIT-III :

Sensors: objectives-robot sensors - sensor classification - micro switches – solid - state switches -proximity sensors - photoelectric sensors - rotary positions - usage and selection of sensors – signal processing - sensors and control integration.

Vision: objectives - Visual sensing - machine Vision - machine vision applications - other optical methods.

UNIT-IV :

Control systems: objectives-control systems correlation – control system requirements – programmable logic controller - PLC programming terminals – proportional – integral – derivative - computer numerical control - microprocessor UNIT - universal robot controller - interfacing - work cell control.

Programming: objectives - robot programming-programming methods - programming languages -levels of robot programming - space position programming - motion interpolation-program statements - sample programs

UNIT-V :

Safety: Objectives - robot safety - safety standards - system reliability-human factor issues – safety sensors and monitoring – safeguarding – training - safety guidelines - definitions.

Industrial Applications: objectives - automation in manufacturing - robot applications – material – handling applications - processing operations – assembly operations - inspection operation - evaluating the potential of a robot application - future applications - challenge for the future – innovations

PRESCRIBED TEXT

- James G.Keramas, *Robot Technology Fundamentals*, Thomson Delmar Publications, 1998.

REFERENCE BOOKS

- Robert J.Schilling, *Fundamentals of Robotics, Analysis & Control*, PHI, 2002.
- Jagannathan Kanniah, M. Fikret Ercan, Carlos A. Acosta Calderon, *Practical Robot Design: Game Playing Robots*
- Saeed B. Niku, *Introduction to Robotics: Analysis, Control, Applications*

****

SEMESTER IV

Programme : M.Sc., **Subject** : Computer Science

Semester: IV **Course** : Digital Image Processing

Course Type : Core XVI **Credits** : 4

Contact Hours: 5 hours/week **Course Code** : MPCSC10

CIA : 25 **CE** : 75

Preamble

This course helps the students to understand fundamental steps in Digital image processing. It also provide knowledge about image compression and image segmentation.

Course Outcomes

On successful completion of the course, the students will be able to

- Recognize the fundamental elements of DIP and representation of an image in multi dimensional aspects
- Apply arithmetic and logical operations on image enhancement process
- Interpret the knowledge on compression techniques for security of an image
- Verify various deduction mechanisms in image segmentation

COURSE CONTENT

UNIT-I :

Introduction: What is Digital Image Processing - Fundamental steps in Digital Image Processing - Components of an Image Processing System.

UNIT-II :

Digital Image Fundamentals: Elements of Visual Perception – Image Sensing and Acquisition - Image Sampling And Quantization – Some Basic Relationship between Pixels – An Introduction to the Mathematical tools used in Digital Image Processing.

UNIT-III :

Intensity Transformation And Spatial Filtering: Background – Some basic Intensity Transformation Functions – Histogram Processing – Fundamentals of Spatial Filtering – Smoothing Spatial Filters.

UNIT-IV :

Filtering In The Frequency Domain: Sampling and the Fourier Transform of Sampled Functions – The Discrete Fourier Transform (DFT) of One Variable – Image Smoothing Using Frequency Domain Filters.

Image Restoration And Reconstruction: A model of the image degradation/ Restoration Process – Restoration in the presence of Noise Only – Spatial Filtering.

UNIT-V :

Color Image Processing: Color Fundamentals – Color Models .

Image Compression: Fundamentals – Some Basic Compression Methods – HUFFMAN Coding – Golomb Coding – Run –Length Coding – Predictive Coding **.**

PRESCRIBED TEXT

- Rafael C Gonzalez, Richard E Woods, *Digital Image Processing*, 3rd Edition, -Pearson Education 2011.

REFERENCE BOOKS

- William K Pratt, Jhon Willey, *Digital Image Processing,* (2001).
- Millman Sonka, Vaclav hlavac, Roger Boyle ,Broos/colic, Thompson Learniy, *Image Processing Analysis and Machine Vision,* (1999)
- A.K.Jain,PHI, *Fundamentals of Digital Image Processing,* New Delhi(1995).

****

Programme : M.Sc.,

Subject : Computer Science

Semester: IV

Course : Grid and Cloud Computing

Course Type : Elective 4.1

Credits : 4

Contact Hours: 5 hours/week

Course Code : MPCSE4

CIA : 25

CE : 75

Preamble

This course helps the students to learn the concepts and architecture of Grid and Cloud Computing. It also covers topics such Grid monitoring, Grid security and Cloud services.

Course Outcomes

On successful completion of the course, the students will be able to

- Understand the basic concepts of Grid and Cloud computing
- Describe the architecture of Grid and Cloud computing
- Acquire knowledge about Grid Scheduling and Cloud Computing services
- Validate Cloud services by using various cloud service providers such as Amazon, Google and Microsoft

COURSE CONTENT

UNIT-I :

Concepts and Architecture: Introduction - Parallel and Distributed Computing - Cluster computing Grid computing - Anatomy and physiology of Grid - Review of web services – OGSA – WSRF

UNIT-II :

Grid Monitoring: Grid Monitoring Architecture (GMA) - An overview of Grid Monitoring systems: Grid ICE – JAMM – MDS - Network Weather Service – Other Monitoring systems: Ganglia and Grid Mon

UNIT-III :

Grid security: Grid Security - A Brief security primer – Cryptography -Grid security – **Grid Scheduling and Resource management:** Scheduling paradigms - Working principles of scheduling - A review of condor ,SGE, PBS and LSF - Grid scheduling with QoS

UNIT-IV :

Examining the Value Proposition: Defining Cloud Computing - Understanding Cloud Architecture - Understanding Services and Applications by Type

UNIT-V :

Using Platforms: Understanding Abstraction and Virtualization - Using Google Web Services - Using Amazon Web Services - Using Microsoft Cloud Services

PRESCRIBED TEXT

- Maozhen Li, Mark Baker, *The Grid: Core Technologies*, John Wiley & Sum 2005
- Barrie Sosinky, *Cloud Computing Bible*, Wiley Publishing Inc, 2011

REFERENCE BOOKS

- Joshy Joseph & Craig Fellenstein, *Grid Computing*, Pearson Education, 2004
- Lars Nielsen, *The Little Book of Cloud Computing*, 2013 Edition.

****

Programme : M.Sc.,
Subject : Computer Science
Semester: IV
Course : Internet of Things
Course Type : Elective 4.2
Credits : 4
Contact Hours: 5 hours/week
Course Code : MPCSE4
CIA : 25
CE : 75

Preamble

This course introduces the fundamentals of Internet of Things and required protocols. It also enables the students to apply the concepts of IoT in the real world scenario.

Course Outcomes

On successful completion of the course, the students will be able to

- Understand the basic concepts of IoT
- Classify the various IoT Architecture
- Analyze various Protocols of IoT
- Construct portable IoT with RASPBERRY Pi and ARDUINO
- Deploy an IoT application and connect to the cloud

COURSE CONTENT

UNIT-I :

Introduction to IoT: Internet of Things – Physical Design – Logical Design – IoT Enabling Technologies – IoT Levels & Deployment Templates – Domain Specific IoTs – IoT and M2M – IoT System Management with NETCONF – YANG – IoT Platforms Design Methodology.

UNIT-II :

IoT Architecture: M2M high-level ETSI Architecture – IETF Architecture for IoT – OGC Architecture – IoT Reference model – Domain model -information model - functional model – communication model - IoT Reference Architecture.

UNIT-III :

IoT Protocols: Protocol Standardization for IoT – Efforts – M2M and WSN Protocols - SCADA and RFID Protocols – Unified Data Standards - Protocols – IEEE 802.15.4 – BACNet Protocol - Modbus – Zigbee Architecture - Network Layer – 6LowPAN – CoAP – Security

UNIT-IV :

Building IoT with RASPBERRY Pi and ARDUINO: Building IoT with RASPBERRY Pi – IoT Systems – Logical Design using Python – IoT Physical Devices and Endpoints – IoT Device – Building blocks – Raspberry Pi – Board – Linux on Raspberry Pi – Raspberry Pi Interfaces - Programming Raspberry Pi with Python – Other IoT Platforms - Arduino

UNIT-V :

IoT Design: Home Automation, Cities, Environment, Agriculture, Productivity Applications.

Real world design constraints: Applications – Asset management, Industrial automation, smart grid ,Commercial building automation ,Smart cities – participatory sensing – Data Analytics for IoT – Software & Management Tools for IoT Cloud Storage Models & Communication APIs – Cloud for IoT – Amazon Web Services for IoT

PRESCRIBED TEXT

- Arshdeep Bahga , Vijay Madisetti, *Internet of Things –A hands –on Approach*, Universities Press 2015

REFERENCE BOOKS

- Dieter Uckelmann , Mark Harrison, Michahelles,Florian(Eds), *Architecting the Internet of Things*, Springer,2011.
- Honbo Zhou, *The Internet of Things in the cloud: A Middleware Perspective*, CRC Press,2012.
- Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stamatis , Karnouskos Stefan Avesand , David Boyle, *From Machine – to- Machine to the Internet of Things – Introduction to a New Age of Intelligence*. Elsevier 2014.
- Olivier Hersent, David Boswarthick ,Omar Elloumi, *The Internet of things – Key applications and Protocols* ,Wiley 2012.

****

Programme : M.Sc.,
Subject : Computer Science
Semester: IV
Course : Soft Computing
Course Type : Elective 4.3
Credits : 4
Contact Hours: 5 hours/week
Course Code : MPCSE4
CIA : 25
CE : 75

Preamble

This course offers knowledge about neural network, basis of Fuzzy logic, fuzzy relations, fuzzy inference system and defuzzification techniques.

Course Outcomes

On successful completion of the course, the students will be able to

- Discuss the nature of soft computing and its applications
- Apply soft computing techniques in small applications
- Analyze various soft computing techniques to solve real life problems
- Evaluate the basis of Fuzzy logic, fuzzy relations and defuzzification techniques

COURSE CONTENT

UNIT-I :

Neural Networks : Fundamentals of Neural Networks – Basic Concepts of Neural Networks – Model of an Artificial Neuron – Neural Network Architecture – Characteristics of Neural Network – Learning Methods – Taxonomy of Neural Network Architecture – Back Propagation Network – Architecture of Back Propagation Network – Back Propagation Learning

UNIT-II :

Neural Network Associative Memory: Auto Correlations – Hetero Correlations – Exponential BAM – Associative Memory for Real Coded Pattern Pairs – Adaptive Resonance Theory – Introduction – ART1 – ART 2 - Applications

UNIT-III :

Fuzzy Set Theory: Crisp Sets – Fuzzy Sets – Crisp Relations – Fuzzy Relations – Fuzzy Systems: Crisp Logic – Predicate Logic – Fuzzy Logic – Fuzzy Rule Based System – Defuzzification Method - Applications

UNIT-IV :

Genetic Algorithms: History – Basic Concepts – Creation of off Springs – Working Principle – Encoding – Fitness Function – Reproduction .Genetic Modeling – Inheritance Operators – Cross Over – Inversion and Deletion – Mutation Operator –Applications – Advances in Genetic Algorithm

UNIT-V :

Hybrid System: Integration of Neural Network – Fuzzy Logic – Genetic Algorithm-Hybrid System – Neural Network – Fuzzy Logic – Genetic Algorithm Weight Determination – Application – Fuzzy Back Propagation Network – Language Recognition Type Fuzzy Members – Fuzzy Neuron – Fuzzy Back Propagation Architecture – Learning in Fuzzy Back Propagation – Applications – Knowledge Base Evaluation

PRESCRIBED TEXT

- S.Rajasekaran and G.A.Vijayalakshmi Pai, *Neural Networks,Fuzzy Logic and Genetic Algorithms Synthesis and Application*, Prentice Hall of India,Pvt. Ltd. (2011)

REFERENCE BOOKS

- Vinoth Kumar and R. Saravana Kumar, *Neural Network and Fuzzy logic*, S.K. Katria & Sons (2012)
- Haykin Simon, *Neural Networks and Learning Machines*, 3/e, Prentice Hall of India (2011)
- Tang,Tan and Yi, *Neural Networks: Computational Models and Application*, Springer Verlag Publications, (2010)

****

Programme : M.Sc.,
Subject : Computer Science
Semester: IV
Course : Project Work
Course Type : Core XVII
Credits : 10
Contact Hours: 20 hours/week
Course Code : MPCSE4
CIA : 25
CE : 75

Preamble

The objective of the project is to motivate the students to work in emerging/latest technologies, help the students to develop ability, to apply theoretical and practical tools/techniques to solve real life problems related to industry, academic institutions and research laboratories.

Course Outcomes

On successful completion of the course, the students will be able to

- Understand the problem.
- Implement & execute the real time application.
- Analyze various testing methods.
- Verify the expected results in real time applications.

COURSE CONTENT

The project is of 20 hours/week for one (semester IV) semester duration and a student is expected to do planning, analyzing, designing, coding, and implementing the project. The initiation of project should be with the project proposal. The synopsis approval will be given by the project guides.

The project proposal should include the following:

- Title
- Objectives
- Input and output
- Details of modules and process logic
- Data Flow Diagram
- Limitations of the project
- Tools/platforms, Languages to be used
- Scope of future applications

****