Scheme of Examination

Second Semester
Master of Computer Application
(wef-2020-21)

S.No	Subject Code	Subject Name	Periods per week			Credits	Maximum Marks (Theory Slot)			Maximum Marks (Practical Slot)		Total Marks
			L	T	P		End sem. Exam	Tests (Two)	Assignment /Quiz	End Sem Practical / Viva	Practical record/ Assignment/Quiz/ Presentation	
1	CA 201	Artificial Intelligence & Applications	3	1	-	4	70	20	10	-	-	100
2	CA202	Mobile Computing	3	1	-	4	70	20	10	-	-	100
3	CA203	Computer Graphics	3	1	-	4	70	20	10	-	-	100
4	CA 204	Design & Analysis of Algorithms	3	1	-	4	70	20	10	-	-	100
5	CA 205	Java Programming & Technologies	3	1	-	4	70	20	10	-	-	100
6	CA 206	Minor Project -I	-	-	8	8	-	-	-	120	80	200
7	CA 207	Programming Lab In Java	-	-	2	2	-	-	-	30	20	50
		Total	**15**	**5**	**10**	**30**	**350**	**100**	**50**	**150**	**100**	**750**

L: Lecture - T: Tutorial - P: Practical

CA-201 Artificial Intelligence & Applications

Objectives:

The primary objective of this course is to introduce the basic principles, techniques, and applications of Artificial Intelligence. Emphasis will be placed on the teaching of these fundamentals, not on providing a mastery of specific software tools or programming environments. Assigned projects promote a 'hands-on' approach for understanding, as well as a challenging avenue for exploration and creativity. Specifically:

1. Gain a historical perspective of AI and its foundations.
2. Become familiar with basic principles of AI toward problem solving, inference, perception, knowledge representation, and learning.
3. Investigate applications of AI techniques in intelligent agents, expert systems, artificial neural networks and other machine learning models.
4. Experience AI development tools such as an 'AI language', expert system shell, and/or data mining tool.
5. Experiment with a machine learning model for simulation and analysis.
6. Explore the current scope, potential, limitations, and implications of intelligent systems.

Total (60 Hours)

Unit-I (12 Hours)

General Issues and Overview of AI

The AI problems, what is an AI technique, Characteristics of AI applications. Introduction to LISP Programming: Syntax and numeric functions, Basic list manipulation functions, predicates and Conditionals, input output and local variables, iteraction and recursion, property lists and arrays

Unit-II (12 Hours)

Problem Solving, Search and Control Strategies

General problem solving, production systems, control strategies forward and backward chaining, Exhausive searches depth first breadth first search.

Heuristic Search Techniques

Hill climbing, branch and bound technique, best first search & A* algorithm, AND / OR graphs, problem Reduction & AO* algorithm, constraint satisfaction problems.

Unit-III (12 Hours)

Knowlede Representations

First order predicate calculus, skolemization, resolution principle & unification, interface mechanisms, Horn's clauses, semantic networks, frame systems and value inheritance, scripts, conceptual dependency.

Unit-IV (12 Hours)

Natural Language processing: Parsing techniques, context free grammer, recursive transitions nets (RNT),augmented transition nets (ATN), case and logic grammers, symantic analysis.
Game playing Minimax search procedure, alpha-beta cutoffs, additional refinments.
Planning Overview an example domain the block word, component of planning systems, goal stack planning, non linear planning.

Unit-V (12 Hours)

Probabilistic Reasoning and Uncertainty: Probability theory, bayes theorem and bayesian networks, Certainty factor. **Expert Systems** Introduction to expert system and application of expert systems, various expert system shells, vidwan frame work, knowledge acquisition, case studies, MYCIN. **Learning** Rote learning, learning by induction, explanation based learning.

Course outcomes:

Upon successful completion of this course, the student shall be able to:
1) Demonstrate fundamental understanding of the history of artificial intelligence (AI) and its foundations.
2) Apply basic principles of AI in solutions that require problem solving, inference, perception, knowledge representation, and learning.
3) Demonstrate awareness and a fundamental understanding of various applications of AI techniques in intelligent agents, expert systems, artificial neural networks and other machine learning models.
4) Demonstrate profciency developing applications in an 'AI language', expert system shell, or data mining tool.
5) Demonstrate profciency in applying scientifc method to models of machine learning.
6) Demonstrate an ability to share in discussions of AI, its current scope and limitations, and societal implications.

Books

1. Elaine Rich and Kevin Knight “Artifical Intelligence” - Tata McGraw Hill.
2. Dan W. Patterson “Introduction to Artifical Intelligence and Expert Systems”, Prentice India.
3. Nils J. Nilson “Principles of Artifical Intelligence”, Narosa Publishing House.

CA-202 Mobile Computing

Course Objectvive:-

1.This course introduces the basic concepts and principles in mobile computing.

2. This includes the major techniques involved, and networks & systems issues for the design and implementation of mobile computing systems and applications.

3. This course also provides an opportunity for students to understand the key components and technologies involved and to gain hands-on experiences in building mobile applications.

Total-(60 Hours)

Unit-I (12 Hours)

Overview of OSI Model: Significance of layered Model, PDUs, SDUs,IDUs, Higher layer Protocols. Switching and Components. Introduction, Applications, history, of wired & wireless Communication Systems. Radio Transmission: frequencies ,signal propagation, antenna, types of modulation, FHSS, DSSS.Multiple Access technology for Wireless Communication: FDMA, TDMA, CDMA Cellular System: Introduction, types.

Unit-II (12 Hours)

Mobile Data Communication: Cellular Telephony, Structure, Fading, Small scale fading, Multi-path Fading, Speech Coding, Error Coding and Correction, Hand off Management, Switching and authentication, MTSO interconnections, frequency hopping, frequency reuse. Circuit Switched Data Services & Packet Switched Data Services on Cellular Networks, Personal Communication Systems (PCS) Architecture, Digital Enhanced Cordless Telecommunications (DECT,) Personal Access Comm. System (PACS).

Unit-III (12 Hours)

Digital Cellular Systems and Standards: GSM System overview, Architecture, GSM Protocol Model, GSM Mobility Management, SMS security aspects. Broadcast System overview. General Packet Service (GRPS)Architecture, GRPS Network, Interfaces and Procedures (2.5 G), 3G Mobile Services: UMTS and International Mobile Telecommunications (IMT-2000), W-C DMA and CDMA 2000,Quality of service in3G.

Unit-IV (12 Hours)

WLAN: Components and working of Wireless LAN, Transmission Media for WLAN, Infrastructure & Types of WLAN, IEEE 802.11 Standards, Protocols for WLAN, MACA, MACAW, Infrared technology. Wireless Application Protocal(WAP) model, architecture, Gateway, WAP protocols and WML.

Unit-V (12 Hours)

Introduction to Bluetooth technology, Wireless in local loop(WLL)architecture, products, Satellite as a switch, Components of VSAT system, VSAT topologies, access schemes Special topics: Wireless and Mobile Computation SS7, GSM, CDMA, Mobile IP, Wireless Mobile ATM, Multicast Routing Protocols, Location Management, Mobile Agents, Mobility Management.

Learning Outcomes:-

1. Describe the basic concepts and principles in mobile computing
2. Understand the concept of Wireless LANs, PAN, Mobile Networks, and Sensor Networks
3. Explain the structure and components for Mobile IP and Mobility Management
4. Understand positioning techniques and location-based services and applications
5. Describe the important issues and concerns on security and privacy

Books

1. Jochen Schiller “Mobile Communication”,Pearson Education.
2. Raj Panadaya “Mobile and Personal Communication System & services
3. Lee “Mobile Cellular Telecom” 1995 Mc Graw Hill

CA-203 Computer Graphics

Course Objectives:

1. To introduce the use of the components of a graphics system and become familiar with building approach of graphics system components and algorithms related with them.

2. To learn the basic principles of 3- dimensional computer graphics.

3. Provide an understanding of how to scan convert the basic geometrical primitives, how to transform the shapes to fit them as per the picture definition.

4. Provide an understanding of mapping from a world coordinates to device coordinates, clipping, and projections.

5. To be able to discuss the application of computer graphics concepts in the development of computer games, information visualization, and business applications.

6. To comprehend and analyze the fundamentals of animation, virtual reality, underlying technologies, principles, and applications.

Total - (60 Hours)

Unit-I (12 Hours)

What is Graphics, Application of Graphics, Elements of Graphics Workstation, Graphics I/P Devices-Keyboard, Trackball, Joystick, Light Pen, Digitizing Tables, Mouse, Touch Panels, Image Scanners . Graphics Display Devices-Raster Scan System, Random Scan System, Arch Of Vector and Raster Scan Display, Refresh CRT, Gray S Hade.

Unit-II (12 Hours)

DRAWING GEOMETRY: Point – Plotting, Coordinate System, Point Plotting, Line Drawing –Line Segments, Line Drawing Algo : DDA Algo, Bresenham's Line Algorithm. Circle Drawing Polygon Representation Ellipse, Rectangle, Filling – Filled Area Primitives, Scan Line Polygon Fill Algo, Flood Fill Algo, Boundary Fill Algorithm

Unit-III (12 Hours)

2D Geometric Transformation : Translation, Rotation, Scaling, Geometric Transformation, Coordinate Transform and Composite Transformation, 2D Viewing Transformation & Clipping : World Coordinate System (WCS), Normalized Device Coordinate System , Windows Viewing View Ports Viewing, Point Clipping, Line Segment Clipping, Coahen – Sutherland, Line Clipping, Polygon Clipping.

Unit-IV (12 Hours)

3D Geometric Transformation 3D Geometric Transformation: Translation, Rotation, Scaling, Coordinate Transform Geometric Transformation Composite Transformation, 3D Display Methods – Parallel Projection, Perspective Projection 3D Viewing & Clipping.

Unit-V (12 Hours)

Segment, Segment Table, Segment Creation, Deletion, Closing, Renaming, Curve Generation , B – Spline Curves, Bezier Curves, Hidden Surface, Z – Buffer Algorithm, Scan Line Algorithm, Painters Algorithm, Depth Comparisons.

Course outcomes:

Students will able to:

1. To list the basic concepts used in computer graphics.
2. To implement various algorithms to scan, convert the basic geometrical primitives, transformations, Area filling, clipping.
3. To describe the importance of viewing and projections.
4. To define the fundamentals of animation, virtual reality and its related technologies.
5. To understand a typical graphics pipeline
6. To design an application with the principles of virtual reality.

Books

1. D.Hearn and M.P. Baker "Computer Graphics" (2nd ed), PHI.
2. COMPUTER GRAPHICS : SCHAUM'S OUTLINE SERIES
3. Roger S. David "Procedural Elements for Computer Graphics", McGraw Hill.

CA-204 Design and Analysis of Algorithms

Objectives of the course

1. Analyze the asymptotic performance of algorithms.
2. Write rigorous correctness proofs for algorithms.
3. Demonstrate a familiarity with major algorithms and data structures.
4. Apply important algorithmic design paradigms and methods of analysis.
5. Synthesize efficient algorithms in common engineering design situations.

Total- (60 Hours)

UNIT-I (12 Hours)

Pre-requisites: Data structure & Discrete structures, models of computation, algorithm analysis, order Architecture, time space complexities average and worst case analysis.

UNIT-II (12 Hours)

Divide and conquer: Structure of divide-and-conquer algorithms: examples; Binary search, quick sort, Strassen Multiplication; Analysis of divide and conquer run time recurrence relations.
Graph searching and Traversal: Overview, Traversal methods (depth first and breadth first search)

UNIT-III (12 Hours)

Greedy Method: Overview of the greedy paradigm examples of exact optimization solution (minimum Cost spanning tree), Approximate solution (Knapsack problem), Single source shortest paths.
Branch and bound: LC searching Bounding, FIFO branch and bound, LC branch and bound application: 0/1 Knapsack problem, Traveling Salesman Problem, searching & sorting algorithms.

UNIT-IV (12 Hours)

Dynamic programming: Overview, difference between dynamic programming and divide and conquer, Applications: Shortest path in graph, Matrix multiplication, Traveling salesman Problem, longest Common sequence.
Back tracking: Overview, 8-queen problem, and Knapsack problem

UNIT-V (12 Hours)

Computational Complexity: Complexity measures, Polynomial Vs non-polynomial time complexity; NP-hard and NP-complete classes, examples.
Combinational algorithms, string processing algorithm, Algebric algorithms, set algorithms

Course Outcomes

1. For a given algorithms analyze worst-case running times of algorithms based on asymptotic analysis and justify the correctness of algorithms .
2. Describe the greedy paradigm and explain when an algorithmic design situation calls for it. For a given problem develop the greedy algorithms.
3. Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it. Synthesize divide-and-conquer algorithms. Derive and solve recurrence relation.
4. Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it. For a given problems of dynamic-programming and develop the dynamic programming algorithms, and analyze it to determine its computational complexity.
5. For a given model engineering problem model it using graph and write the corresponding algorithm to solve the problems.
6. Explain the ways to analyze randomized algorithms (expected running time,

probability of error).

7. Explain what an approximation algorithm is. Compute the approximation factor of an approximation algorithm (PTAS and FPTAS).

BOOKS:

1. Ullman "Analysis and Design of Algorithm" TMH
2. Sara Basse, A. V. Gelder, " Computer Algorithms," Addison Wesley
3. T. H. Cormen, Leiserson , Rivest and Stein, "Introduction of Computer algorithm," PHI
4. E. Horowitz, S. Sahni, and S. Rajsekaran, "Fundamentals of Computer Algorithms," Galgotia Publication

CA-205 Java Programming & Technologies

Course 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.
5. Be able to use the Java SDK environment to create, debug and run simple Java programs.

Total-(60 Hours)

UNIT-I (12 Hours)

The Java Environment: History of Java: Comparison of Java and C++; Java as an object oriented language: Java buzzwords; A simple program, its compilation and execution; the concept of CLASSPATH; Basic idea of application and applet;

Basics: Data types; Operators- precedence and associativity; Type conversion; The decision making – if, if ..else, switch; loops – for, while, do…while; special statements–return, break, continue, labeled break, labeled continue; Modular programming methods; arrays; memory allocation and garbage collection in java keywords.

Object Oriented Programming in Java: Class; Packages; scope and lifetime; Access specifies; Constructors; Copy constructor; this pointer; finalize () method; arrays; Memory allocation and garbage collection in java keywords

Inheritance : Inheritance basics, method overriding, dynamics method dispatch, abstract classes.

UNIT-II (12 Hours)

Interfaces : defining an interface, implementing & applying interfaces, variables in interfaces, extending interfaces.

Multithreading and Exception Handling: Basic idea of multithreaded programming; The lifecycle of a thread; Creating thread with the thread class and runnable interface; Thread synchronization; Thread scheduling; Producer-consumer relationship; Daemon thread, Selfish threads; Basic idea of exception handling; The try, catch and throw; throws Constructor and finalizers in exception handling; Exception Handling.

UNIT-III (12 Hours)

Applets: Applet security restrictions; the class hierarchy for applets; Life cycle of applet; HTML Tags for applet.

The AWT: The class hierarchy of window fundamentals; The basic user interface components Label, Button, Check Box, Radio Button, Choice menu, Text area, Scroll list, Scroll bar; Frame; Layout managersflow
layout, Grid layout, Border layout, Card layout.

The Java Event Handling Model: Java's event delegation model – lgnoring the event, self contained events, Delegating events; The event class hierarchy; The relationship between interface, methods called, parameters and event source; Adapter classes; Event classes action Event, Adjustment Event, Container Event, Focus Event, Item Event, Eey Event, Mouse Event, Text Event, Window Event.

UNIT-IV (12 Hours)

Input/output: Exploring Java i.o., Directories, stream classes
The Byte stream: Input stream, output stream, file input stream, file output stream, print stream, Random access file, the character streams, Buffered reader, buffered writer, print writer, serialization.

JDBC: JDBC-ODBC bridge; The connectivity model; The driver manager; Navigating the resultset object contents; java.sql Package; The JDBC exception classes; Connecting to Remote database.

UNIT-V (12 Hours)
Networking & RMI: Java Networking: Networking Basics: Socket, Client server, reserved sockets, proxy Servers, Inet address, TCP sockets, UDP sockets.
; RMI for distributed computing; RMI registry services; Steps of creating RMI Application and an example.
Collections: The collections framework, collection interfaces, collection classes.

Course outcomes

1. Knowledge of the structure and model of the Java programming language, (knowledge)
2. Use the Java programming language for various programming technologies (understanding)
3. Develop software in the Java programming language, (application)

BOOKS

1. Naughton & Schildt "The Complete Reference Java 2", Tata McGraw Hill
2. Deitel "Java- How to Program:" Pearson Education, Asia
3. Horstmann & Cornell "Core Java 2" (Vol I & II), Sun Microsystems
4. lvan Bayross "Java 2.0" : BPB publications
5. Ivor Horton's "Beginning Java 2, JDK 5 Ed., Wiley India.

List of Program to be perform (Expandable)

1. Installation of J2SDK
2. Write a program to show Scope of Variables
3. Write a program to show Concept of CLASS in JAVA
4. Write a program to show Type Casting in JAVA
5. Write a program to show How Exception Handling is in JAVA
6. Write a Program to show Inheritance
7. Write a program to show Polymorphism
8. Write a program to show Access Specifiers (Public, Private, Protected) in JAVA
9. Write a program to show use and Advantages of CONTRUCTOR
10. Write a program to show Interfacing between two classes
11. Write a program to Add a Class to a Package
12. Write a program to show Life Cycle of a Thread
13. Write a program to demonstrate AWT.
14. Write a program to Hide a Class
15. Write a Program to show Data Base Connectivity Using JAVA
16. Write a Program to show "HELLO JAVA " in Explorer using Applet
17. Write a Program to show Connectivity using JDBC
18. Write a program to demonstrate multithreading using Java.
19. Write a program to demonstrate applet life cycle.
    Write a program to demonstrate concept of servlet.