

Chhattisgarh Swami Vivekanand Technical University Bhilai(C.G.)

SCHEME OF TEACHING & EXAMINATION

Master of Computer Applications (MCA) – IV Semester

S.No.	Board of Study	Subject Code	SUBJECT	Period Per Week			Scheme of Exam			Total Marks	Credit L+(T+P)/2
				L	T	P	Theory / Practical				
							ESC	CT	TA		
1	Computer Applications	521451(21)	Artificial Intelligence and Expert System	4	1	0	100	20	20	140	5
2	Computer Applications	521452(21)	Compiler Design	4	1	0	100	20	20	140	5
3	Computer Applications	521453(21)	Software Engineering	4	1	0	100	20	20	140	5
4	Computer Applications	521454(21)	Advanced JAVA Programming	4	1	0	100	20	20	140	5
5	<i>Refer Table-I</i>		<i>Elective-I</i>	4	1	0	100	20	20	140	5
6	Computer Applications	521461(21)	Artificial Intelligence Lab	0	0	4	50	–	25	75	2
7	Computer Applications	521462(21)	Advanced Java Lab	0	0	4	75	–	25	100	2
8	Computer Applications	521463(21)	Project-I	0	0	5	75	–	25	100	3
9	Computer Applications	521464(21)	Aptitude Training	0	0	2	–	–	25	25	1
TOTAL				20	5	15	700	100	200	1000	33

L: Lecture **T:** Tutorial **P:** Practical
ESE: End Semester Examination **CT:** Class Test **TA:** Teacher's Assessment

Table 1 : List of Elective -I

S.No.	Board of Study	Code	Subject
1	Computer Applications	521471(21)	Neural Network & Fuzzy Logic
2	Computer Applications	521472(21)	Logic & Functional Programming
3	Computer Applications	521473(21)	Network Programming
4	Computer Applications	521474(21)	Modeling & Simulation
5	Computer Applications	521475(21)	Design & Analysis of Algorithms
6	Computer Applications	521476(21)	System Programming
7	Computer Applications	521477(21)	Distributed Operating System
8	Computer Applications	521478(21)	Cloud Computing
9	Computer Applications	521479(21)	Biometric Security

Note:

- 1/4th of total strength of students subject to Minimum Strength of twenty students is required to offer an elective in the college in a particular academic session.
- Choice of elective course once made for an examination cannot be changed for future examinations.

Name of the Programme: Master of Computer Applications (MCA) ::: Duration of the Programme: Three years

Chhattisgarh Swami Vivekanand Technical University, Bhilai

Name of the programme: **Master of Computer Applications (MCA)**

Semester: **IV**

Subject: **Artificial Intelligence & Expert Systems**

Code: **521451 (21)**

Total Theory Periods: **40**

Total Tutorial **10**

Periods:

Class Test: **Two (Minimum)**

Assignments: **Two (Minimum)**

Duration of ESE: **Three Hours**

Maximum Marks in ESE: **100**

Minimum Marks in ESE: **40**

Course objectives:

1. To make students learn to define problem of complex nature , state space of problem domain and searching techniques to solve them.
2. To make students understand concept of *heuristic* and how it is applied to solve AI based problem along with mechanism to represent knowledge structures and inference procedure.
3. To make student learn processing of natural language and challenges associated with it.
4. To make students understand concept of Expert System, its design issues and applications
5. To make students aware with fundamental programming environment in PROLOG and LISP.

Course Outcomes:

- a) Student will have ability to understand and define different AI problem and apply suitable problem solving technique.
- b) Student will have ability to define the heuristics and apply them for solving complex problem with understanding of different heuristic based search techniques.
- c) Student will develop an understanding of game playing techniques
- d) Student will have understanding of different knowledge structure and inference mechanism with ability to apply them in intelligent solutions of complex problem .
- e) Students will develop skills needed for processing of natural language at syntactic and semantic level.
- f) Student will understand the existence of uncertainty in problem solving and how mathematical /statistical models are used to overcome these problems.
- g) Students will understand planning system and different types of planning required for problem solving process
- h) Student will be able to understand working of Expert system.
- i) Student will develop skill to make program in PROLOG and LISP.

UNIT I General overview of AI & Search techniques: Introduction to AI, The AI problems. Characteristics of AI applications, Problem Solving, Search and Control Strategies General Problem solving; Production systems, Blind searches: Depth first, Breadth first search. Informed Search: Hill climbing; Branch and Bound technique; Best first search, Constraint Satisfaction problems.

UNIT II Heuristic Search techniques & Knowledge Representation: A* algorithm, AND/OR Graphs; Problem reduction and AO* algorithm; Game Playing: Minimax search procedure; Alpha-Beta cutoffs; First Order Predicate Calculus; Skolemisation; Resolution Principle and Unification Algorithm; Semantic Networks; Frame Systems and Value Inheritance; Scripts; Conceptual Dependency.

UNIT III Natural Language Processing, Planning and Uncertainty: Case and Logic Grammars, Context - free Grammars. Recursive Transition Nets (RTN); Augmented Transition Nets (ATN); Planning Overview - An Example Domain: The Blocks World; Component of Planning Systems; Goal Stack Planning (linear planning); Non-linear Planning using constraint posting ; Probabilistic Reasoning and Uncertainty; Probability theory; Bayes Theorem and Bayesian networks; Certainty Factor.

UNIT IV Expert Systems and Learning: Introduction to Expert Systems, Architecture of Expert Systems; Expert System Shells; Knowledge Acquisition; Case Studies: MYCIN, Learning: Learning model, Types of learning: Rote learning, induction based learning, Winston's learning algorithm, version spaces , Explanation based learning.

UNIT V AI Programming language: Introduction to PROLOG and LISP, Programming techniques, Syntax and Numeric Functions; predicates and conditionals, List manipulation , redundancy and termination, Iteration and Recursion; Property list and Arrays.

Text Book:

1. Elaine Rich and Kevin Knight: Artificial Intelligence- Tata McGraw Hill.
2. Dan W.Patterson, Introduction to Artificial Intelligence and Expert Systems- Prentice Hall of India.
3. B.Yegnanarayana: "Artificial Neural Networks", Prentice Hall of India.
4. SarojKaushik: "Artificial Intelligence", Cenage Learning

Reference Books:

1. Nils J.Nilsson: Principles of Artificial Intelligence- Narosa Publishing house.
2. Artificial Intelligence : A Modern Approach, Stuart Rusell, Peter Norvig, Pearson Education,2nd Edition
3. Artificial Intelligence, Winston, Patrick, Henry, Pearson Education.
4. SimanHaykin, "Neural Netowrks" Prentice Hall of India
5. John Yen, rezaLangari: "Fuzzy Logic" Pearson Education.

Name of the Programme: Master of Computer Applications (MCA) ::: Duration of the Programme: Three years

Chhattisgarh Swami Vivekanand Technical University, Bhilai

Name of the programme: **Master of Computer Applications (MCA)**

Subject: **Compiler Design**

Total Theory Periods: **40**

Class Test: **Two (Minimum)**

Duration of ESE: **Three Hours**

Semester: **IV**

Code: **521452 (21)**

Total Tutorial **10**

Periods:

Assignments: **Two (Minimum)**

Minimum Marks in ESE: **40**

Maximum Marks in ESE: **100**

Course objectives:

1. To introduce various phases of compiler design.
2. To introduce the major concept areas of language translation and compiler design
3. To develop an awareness of the function and complexity of modern compilers.

Course outcomes:

- 1) Students will have a concrete view on the theoretical and practical aspects of compiler design
- 2) Students will be able to apply ideas and techniques discussed to various software design

- UNIT I** **Introduction:** Introduction to Compiler, single and multi-pass compilers, Translators, Phases of Compilers, Compiler writing tools, Bootstrapping, Finite Automata and Lexical Analysis: Role of Lexical Analyzer, Specification of tokens, Recognition of tokens, Regular expression, Finite automata, from regular expression to finite automata, transition diagrams, Implementation of lexical analyzer, Tool for lexical analyzer–LEX, Error reporting.
- UNIT II** **Syntax Analysis and Parsing Techniques:** Context free grammars, Bottom-up parsing and top down parsing, Top down Parsing: elimination of left recursion, recursive descent parsing, Predictive Parsing; Bottom Up Parsing: Operator precedence parsing, LR parsers, Construction of SLR, canonical LR and LALR parsing tables, Construction of SLR parse tables for ambiguous grammar.
- UNIT III** **Syntax Directed Translation & Intermediate code generation:** Synthesized and inherited attributes, dependency graph, Construction of syntax trees, bottom up and top down evaluation of attributes, S-attributed and L-attributed definitions. Postfix notation; Three address code, quadruples, triples and indirect triples, Translation of assignment statements, control flow, Boolean expressions.
- UNIT IV** **Runtime Environment:** Storage organization, activation tree, activation record, allocation strategies: stack and heap, symbol table management, dynamic storage allocation: implicit and explicit.
- UNIT V** **Code Optimization & Code Generation:** Basic blocks and flow graphs, Optimization of basic blocks, Loop optimization, Loop invariant computations. Issues in the design of Code generator, simple Code generator.

TextBooks:

1. Compilers-Principles, Techniques and Tools by Alfred V.Aho, Ravi Sethi and J.D. Ullman, AddisonWesley.
2. PrinciplesofCompilerDesign,AlfredV.AhoandJ.D.Ullman,NarosaPublication.

ReferenceBooks:

1. Compiler design in C by A.C. Holub, Prentice Hallof India.
2. Compiler construction (Theory and Practice) by A. Barret William and M. Bates (Galgotia Publication).
3. Compiler Design, Kakde, Compiler Design, Galgotia Publication.

Chhattisgarh Swami Vivekanand Technical University, Bhilai

Name of the programme:	Master of Computer Applications (MCA)	Semester:	IV
Subject:	Software Engineering	Code:	521453 (21)
Total Theory Periods:	40	Total Tutorial Periods:	10
Class Test:	Two (Minimum)	Assignments:	Two (Minimum)
Duration of ESE:	Three Hours	Maximum Marks in ESE:	100
		Minimum Marks in ESE:	40

Course objectives:

1. To introduce to the students the complexities of large projects, the challenges associated with it and the processes involved in their development and how software engineering alleviates these problems
2. To introduce to the students the importance of good requirement analysis and the formal way of documenting it
3. To train the students to use formal methods for project management and tools for designing systems
4. To teach the students the formal methods used for Testing systems

Course outcomes:

1. The student will have a fair idea about the importance of using software engineering principles in real life projects
2. The student will also be able to pick an appropriate software development model for developing systems
3. The student will be able to prepare software requirement sheet for a real life project, keeping in mind the properties of an SRS document
4. The student will be able to use mathematical models for calculating the size, cost and duration of real life projects
5. The student will be able to test the developed system using different testing techniques

- UNIT I Introduction:** The Software And Software Engineering Problem, Approach And Goals Of Software Engineering. Software Processes And Models: Processes, Projects And Products, Component Software Processes, Characteristics Of A Software Process, Software Development Process, Project Management Process, Software Configuration Management Process. Models: Linear Sequential, Prototyping, Rad, Incremental, Spiral, Winwin Spiral, Concurrent Development Model
- UNIT II Software requirement Analysis and Specification:** Software requirement, Feasibility study, need for SRS, characteristics and component of SRS, specification languages, structured analysis, object oriented modeling, Requirement analysis, DFD, Structure of a requirement document, validation of SRS, requirement reviews, Cost estimation, uncertainties in cost estimation, building cost estimation. Size estimation :- COCOMO model.
- UNIT III Function Oriented design:** Design principles, coupling, cohesion, design notation and specification, structured design technology, verification.**Object Oriented Design :** Overview of Object oriented design, UML diagram, Use CASE diagram, class diagram, interaction diagram (Sequence and collaboration diagram), Activity diagram.
- UNIT IV Software Testing techniques and strategies:** Software testing objectives & principles, test case design, white box testing, black box testing.: A Strategic Approach to software testing, strategic issues, unit, integration testing, validation testing, system testing, object oriented program testing, debugging.
- UNIT V Software Re-engineering:** Software reengineering, software maintenance, a software reengineering process model, reverse engineering, restructuring code, data restructuring, forward engineering, the economics of reengineering.**Computer Aided software Engineering:** What is CASE, building blocks for CASE, taxonomy of CASE tools, integrated CASE environment, the integration architecture, the case repository.**Component Based Software Engineering:** CBSE process, domain engineering, Component based development, economics of CBSE.

Text Books:

1. Pressman Roger, Software Engineering: A Practitioner's Approach TMH, Delhi.
2. JalotePankaj: An Integrated Approach to software Engineering, Narosa, Delhi.

Reference Books:

1. R.E. Fairly, Software Engineering Concepts, McGraw Hill, Inc 1985.
2. Rajib Mall, "Fundamental of Software Engineering", PHI.

Name of the Programme: Master of Computer Applications (MCA) ::: Duration of the Programme: Three years

Chhattisgarh Swami Vivekanand Technical University, Bhilai

Name of the programme:	Master of Computer Applications (MCA)	Semester:	IV
Subject:	Advanced JAVA Programming	Code:	521454 (21)
Total Theory Periods:	40	Total Tutorial Periods:	10
Class Test:	Two (Minimum)	Assignments:	Two (Minimum)
Duration of ESE:	Three Hours	Maximum Marks in ESE:	100
		Minimum Marks in ESE:	40

Course Objectives:

1. To learn the Java programming language fundamentals: its syntax, patterns, and styles.
2. To learn file handling, networking in Java
3. To learn GUI design using Swing components and event handling.
4. To design a web based application using Java servlets and applets.

Course Outcomes:

1. Students will be able to write and test applets for potential inclusion in web pages.
2. Students will be able to understand the use of APIs in robust, enterprise three level application developments.
3. Students will be able to understand the Java features for secure communications over the internet.

- UNIT I Overview of Java:** Features of Java, Byte-code, JVM, data types, variables and arrays, control statements, Introduction to Java class and object, main() function, garbage collection and finalize() method, this, inheritance, method overriding, dynamic method dispatching, super, final, package, interface, abstract class, class path, String and String Buffer Class.
- UNIT II Exception Handling and Multithreading:** Exception types, uncaught Exception, using try- catch, throw, throws, finally, Throwable class and object, Exception classes, create own exception subclass. Creating multiple threads, isAlive(), join(), Thread priorities, synchronization, Deadlock, wait(), notify(), notifyAll() methods, inter-thread communication, suspend, resume and stop the threads. Collection framework - HashSet, ArrayList, HashMap.
- UNIT III Streams and Sockets:** I/O classes & Interfaces, File, The Stream Classes, the Byte stream (InputStream, OutputStream, FileInputStream, FileOutputStream), Serialization. Network basics, Networking classes and Interfaces, InetAddress, TCP/IP Client/Server socket, URL, URLConnection, Datagram, Introduction to RMI.
- UNIT IV Event Handling and Swing:** Delegation event model, event classes, Event listener interface, Layout managers, Swing: benefits of Swing over AWT, JFrames, JPanels, JLabels, JButtons, JTabbedPane, JScrollPane, JSplitPane, JOptionPane, JComboBox, JList, JList, Text components, JMenu, JToolBar, JDialog, JTable, Database connectivity.
- UNIT V Web development:** The Applet class, Applet Architecture, Applet skeleton, HTML APPLET Tag, Passing parameter to Applet, getDocumentBase(), getCodeBase(), Applet Context, showDocument(). Servlet Architecture, Servlet interface, Servlet Request/ Response interface, Servlet designing, using cookies, session management.

Text Books:

1. The Complete Reference Java 2 (Updated to Cover J2SE 1.4), Herbert Scheldt, Tata McGraw-Hill publishing company Ltd. New Delhi, India.
2. Head First Java, [Kathy Sierra](#), [Bert Bates](#), [O'Reilly Media](#).

Reference Books:

1. Java 2 for Professionals Developers, Michael Morgan, SAMS, Techmedia, New Delhi.
2. Thinking in Java, The Definitive Introduction to Object-Oriented Programming in the Language of World-Wide-Web, Bruce Echel, Pearson Education.
3. Core Java 2 Volume-I Fundamentals, Cay S. Horstmann Gary Cornell, Pearson Education.
4. Java 2 Developer's Hand Book, Philip Heller and Simon Roberts, BPB Publication, New Delhi.
5. Java Thread Programming, Paul Hyde, SAMS.
6. Java Swing, Loy and Wood, O' reilly.

Name of the Programme: Master of Computer Applications (MCA) ::: Duration of the Programme: Three years

Chhattisgarh Swami Vivekanand Technical University, Bilai

Name of program: **Master of Computer Applications (MCA)**

Semester: **IV**

Subject: **Artificial Intelligence Laboratory**

Code: **521461 (21)**

Lab Periods/ week: **4**

Batch Size: **30**

Maximum Marks: **50**

Minimum Marks: **25**

List of Experiments:

- (i) Write a prolog program to find the rules for parent, child, male, female, son, daughter, brother, sister, uncle, aunt, ancestor given the facts about father and wife only.
- (ii) Write a program to find the length of a given list
- (iii) Write a program to find the last element of a given list
- (iv) Write a program to delete the first occurrence and also all occurrences of a particular element in a given list.
- (v) Write a program to find union and intersection of two given sets represented as lists.
- (vi) Write a program to read a list at a time and write a list at a time using the well defined read & write functions.
- (vii) Write a program given the knowledge base, If x is on the top of y, y supports x. If x is above y and they are touching each other, x is on top of y. A cup is above a book. The cup is touching that book. Convert the following into wff's, clausal form; Is it possible to deduce that 'The book supports the cup'.
- (viii) Write a program given the knowledge base. If Town x is connected to Town y by highway z and bikes are allowed on z, you can get to y from x by bike. If Town x is connected to y by z then y is also connected to x by z. If you can get to town q from p and also to town r from town q, you can get to town r from town p. Town A is connected to Town B by Road 1. Town B is connected to Town C by Road 2. Town A is connected to Town C by Road 3. Town D is connected to Town E by Road 4. Town D is connected to Town B by Road 5. Bikes are allowed on roads 3, 4, 5. Bikes are only either allowed on Road 1 or on Road 2 every day. Convert the following into wff's, clausal form and deduce that 'One can get to town B from town D'.
- (ix) Solve the classical Water Jug problem of AI.
- (x) Solve the classical Monkey Banana problem of AI.
- (xi) Solve the classical Crypt arithmetic problems such as DONALD + GERALD = ROBERT of AI.
- (xii) Write a program to search any goal using Depth first search.
- (xiii) Write a program to search any goal using Breadth first search.
- (xiv) Solve the classical Missionary Cannibals problem of AI.
- (xv) Solve the classical Travelling Salesman Problem of AI.
- (xvi) Solve the classical Blocks World Problem of AI.
- (xvii) Write a program to search any goal given an input graph using AO* algorithm.
- (xviii) Write a program to solve Tower of Hanoi.
- (xix) Write a program to solve N-Queens problem.
- (xx) Solve the Tic-Tac-Toe Problem of AI.

List of Equipments/Machine required :

- (i) PC with Windows xp
- (ii) Visual prolog compiler

Recommended Books:

- (i) Ivan Bratko : Logic & prolog programming.

Chhattisgarh Swami Vivekanand Technical University, Bhilai

Name of program: **Master of Computer Applications (MCA)**

Semester: **IV**

Subject: **Advanced JAVA Laboratory**

Code: **521462 (21)**

Lab Periods/ week: **4**

Batch Size: **30**

Maximum Marks: **75**

Minimum Marks: **38**

List of experiments:

1. Write a java program that accepts 10-digit string as a telephone number and extracts the 3-digit area code, 3-digit exchange and the remaining 4-digit number as a separate string and print them.
2. Write a Java program to count the number of words in a given sentences.
3. Write a java code to create a 2-D array having 5 rows, with first row having 1 element, second row having 2 elements and so on. Store numbers in these cells by taking input from user and find the sum of the numbers of each row.
4. Write a program that creates an abstract class called Shape. Create two subclasses rectangle and triangle. Include appropriate methods for both the subclasses that calculate and display the area of the rectangle and triangle.
5. Write a java source code to accept elements of an integer array from command line and sort the array.
6. Define an exception NegArgException that is thrown if the argument is negative. Write a program to find the factorial of a number that uses this exception when the number is negative.
7. Create an exception called "NoMatchException" that is thrown if the string is not equal to "India". Handle the exception in you java program.
8. Write Java code to see the all the IP addresses of "www.google.com" using InetAddress class.
9. Write Java code to see the port number, protocol name and host name of the URL <http://download.oracle.com/javase/1.4.2/docs/api/java/lang/String.html>.
10. Write a client server program that sends message to and fro between client and server using UDP protocol.
11. Write a client server program that sends message to and fro between client and server using TCP/IP protocol.
12. Write a java program to read a character file line by line and display its contents with line numbers.
13. Write a java program that display the contents of a directory passed through command line argument.
14. Write a program that illustrates how to use isDirectory() and list() methods to examine the contents of your directory.
15. Write a Java program to copy the contents of one file into other.
16. Create a class called student with roll number and name. Write two objects of student class in a file and print the contents of file.
17. Write a program to create two threads, one thread will print odd numbers and second thread will print even numbers between 1 to 20 numbers.
18. Write a java program that shows that synchronization between producer and consumer.
19. Write an applet program that accepts two input string using <param> tag and concatenate the strings and display it in the applet window.
20. Write a GUI based java program that handles mouse events.
21. Write a GUI based java program that handles key events.
22. Construct a Swing based GUI that performs simple arithmetic operations on two numbers.
23. Write a GUI based java program that copies the contents of one list into other.
24. Write a GUI based java program having the gender in option buttons and the hobbies as check boxes and displays the selections made by the user.
25. Create a simple java Applet that shows the mouse moving position only on the status bar of the Applet window using MouseMotionAdapter class.
26. Write a well-documented Java source code that accepts a string from JOptionPane Input Dialog and verify weather that string is a Palindrome or not.
27. Create a simple servlet used to print "HELLO WORLD".
28. Write a servlet that takes two numbers through the html file and prints the sum of the two numbers.
29. Write Java source code that can transfer the contents of some text file "D:/SampleDir/SouceFile.txt" into another text file "D:/SampleDir/TargetFile.txt" using some appropriate stream class. The code should also consider the possible exceptions.
30. Write a Java code to display all the records of a table stored in MS-Access/ oracle database.

List of Equipments/Machine required:

(i) Software: The Java Development Kit version 1.6 (JDK 1.6 or more) and Java ServletsDevelopment Kit.

Recommended Books:

- (i) "Head First Java" by Kathy Sierra & Bert Bates O'Reilly Publication
- (ii) "Head First Servlets and JSP" Bryan Basham , Kathy Sierra & Bert Bates

Name of the Programme: Master of Computer Applications (MCA) ::: Duration of the Programme: Three years

Chhattisgarh Swami Vivekanand Technical University, Bhilai

Name of program: **Master of Computer Applications (MCA)**

Semester: **IV**

Subject: **Project - I**

Code: **521463 (21)**

Lab Periods/ week: **4**

Batch Size: **30**

Maximum Marks: **75**

Minimum Marks: **38**

Introduction:

A mini-project should be done by the students based on concepts they have already learnt in the first three semesters of the MCA programme. It may be primarily based on database concepts, object oriented concepts, .NET etc.

Guidelines:

1. A student is expected to perform planning, analyzing, designing and implementing the project.
2. The initiation of project should be with the project proposal/Synopsis that is to be treated as an assignment. The synopsis approval will be given by Faculty/Project Guides. Project Proposal should include the following:

Title
Objectives
Input and Output
Process Logic
Limitations of the Project
Tools/Platforms, Languages to be used
Scope of Future Application.

Practical manual format:

The evaluation of the Project - I will be based on the project reports submitted by the student a presentation and a demonstration. **The format of the project Report should be as under:**

1. Abstract
2. List of Tables
3. List of Figures
4. List of Abbreviations/Symbols
5. Project Development Process Model Used (Methodology)
6. Project's Process Documentation
 - 6.1 System Analysis and Design
 - 6.1.1 Initial Description of Problem
 - 6.1.2 Data dictionary
 - 6.1.3 Functional Model
 - 6.1.4 Identification of Input/Output values
 - 6.1.5 DFD as needed to show functional dependencies
 - 6.1.6 Identification of constraint
 - 6.2 Coding Language and Operating System (OS) used (Including explanation)
 - 6.3 Detail Databases Design and Connectivity Procedure
 - E-R Diagram
 - Table Relationship Diagram
 - etc
 - 6.4 Maintenance Criteria
 - 6.5 Developed Project Interfaces and Reports (i.e., I/O Interface)
 - 6.6 Features of Project
 - 6.7 Future Enhancement scope of the project
7. Summary and Conclusions
8. Reference/Bibliography
9. Short questions for each experiment:
 - What is the aim of the project?
 - Why the particular software is used?
 - What are the limitations and scope of improvement of your project?
 - Explain the source of data collection and its reliability?
 - What was the importance of analysis and design in your project?List of equipment / machines / instruments / tools / software, if any:
10. The student can develop the project using any of the following -
 - i. VB.NET or C#
 - ii. C Language
 - iii. C++ Language
 - iv. ASP.NET

Name of the Programme: Master of Computer Applications (MCA) ::: Duration of the Programme: Three years

Chhattisgarh Swami Vivekanand Technical University, Bhilai

Name of program: **Master of Computer Applications (MCA)**

Semester: **IV**

Subject: **Aptitude Training**

Code: **521464 (21)**

Lecture Hours: **2 per week**

TA Marks: **25**

Course Objectives:

1. To learn the basic formula to solve the problem of aptitude.
2. To learn the technique to solve the problem of aptitude.

Course Outcomes:

1. Students will be able to solve the aptitude and reasoning problem.
2. Students will be able to prepare for competitive exam .

UNIT I Arithmetical Ability: Number, Average, Problems on Age, Percentage, Profit & Loss, Simple Interest, Compound Interest, Ratio & Proportion, Partnership.

UNIT II Arithmetical Ability: Chain Rule, Time & Work, Time & Distance, Allegation or Mixtures, Area, Volume & Surface Areas, Permutation & Combination, Probability.

UNIT III Reasoning: Classification, Series Completion, Coding-Decoding, Blood Relation, Puzzle Test, Alphabet Test, Mathematical Operations, Logical Sequence of Words, Sequential Output Tracing, Direction Sense Test, Number Ranking & Time Sequence Test.

UNIT IV Logical Reasoning: Logic (Syllogism), Statement-Arguments, Course of Action, Conclusions from Passages. Miscellaneous Logical Puzzles.

UNIT V Test Series

Text Books:

1. Quantitative Aptitude By R S Agrawal, S Chand Publication.
2. Verbal & Nonverbal Reasoning By R S Agrawal S Chand Publicatio.

Reference Books

1. Quantitative Aptitude and Reasoning By Praveen, R.V., PHI.

Chhattisgarh Swami Vivekanand Technical University, Bhilai

Name of the programme: **Master of Computer Applications (MCA)**
Subject: **Neural Network & Fuzzy Logic**
(Elective – I)

Semester: **IV**
Code: **521471 (21)**

Total Theory Periods: **40**

Total Tutorial **10**
Periods:

Class Test: **Two (Minimum)**

Assignments: **Two (Minimum)**

Duration of ESE: **Three Hours**

Maximum Marks in ESE: **100**

Minimum Marks in ESE: **40**

Course Objectives:

1. The main objective of this course is to provide the student with the basic understanding of neural networks and fuzzy logic fundamentals, Program the related algorithms and Design the required and related systems.
2. To learn the various architectures of building an ANN and its applications
3. To learn the advanced methods of representing information in ANN like self organizing networks, associative and competitive learning
4. To learn the fundamentals of Crisp sets, Fuzzy sets and Fuzzy Relations.

Course Outcome:

1. Students will be able to understand Artificial Neural Network concept with the help of Biological Neural Network
2. Students will be able to implement algorithms to train ANN by using learning algorithms
3. Students will be able to test fuzzy set operations and binary relations

UNIT I Introduction to Artificial Neural Networks: Elementary Neurophysiology Models of a Neuron, Neural Networks viewed as directed graphs, Feedback, from neurons to ANN, Artificial Intelligence and Neural Networks; Network Architectures, Singlelayered Feed forward Networks, Multi-layered Feed-forward Networks, Recurrent Networks, Topologies.

UNIT II Learning and Training :Activation and Synaptic Dynamics, Hebbian, Memory based, Competitive, Error-Correction Learning, Credit Assignment Problem: Supervised and Unsupervised learning, Memory models, Stability and Convergence, Recall and Adaptation.

UNIT III A Survey of Neural Network Models :Single-layered Perceptron – least mean square algorithm, Multi-layered Perceptrons – Back propagation Algorithm, XOR – Problem,The generalized Delta rule, BPN Applications, Adalines and Madalines – Algorithm and applications.

UNIT IV Applications :Talking Network and Phonetic typewriter : Speech Generation and Speech recognition, Neocognitron - Character Recognition and Handwritten Digit recognition, Pattern Recognition Applications.

UNIT V Neural Fuzzy Systems:Introduction to Fuzzy sets, operations, relations, Examples of Fuzzy logic, Defuzzification, Fuzzy Associative memories, Fuzziness in neural networksexamples, Fuzzy Rules and Fuzzy Reasoning : Extension Principles and Fuzzy Relations, Fuzzy IF THEN Rules, Fuzzy Reasoning

Text Books:

1. Artificial Neural Networks by B. Yagna Narayan, PHI
2. Neural Network Design by Hagan Demuth Deale Vikas Publication House.

Reference Books:

1. Neural Networks by James A. Freeman and David M. Strapetus, Pearson Education.
2. Neural Network & Fuzzy System by Bart Kosko, PHI.

Name of the Programme: Master of Computer Applications (MCA) ::: Duration of the Programme: Three years

Chhattisgarh Swami Vivekanand Technical University, Bhilai

Name of the programme: **Master of Computer Applications (MCA)**
Subject: **Logic & Functional Programming**
(Elective – I)

Semester: **IV**
Code: **521472 (21)**

Total Theory Periods: **40**

Total Tutorial **10**
Periods:

Class Test: **Two (Minimum)**

Assignments: **Two (Minimum)**

Duration of ESE: **Three Hours**

Maximum Marks in ESE: **100**

Minimum Marks in ESE: **40**

Course Objectives:

1. Objective of the course is to introduce applicative/declarative style of computing based on mathematical principles, as opposed to the imperative style.
2. There are mainly two computing paradigms in the applicative/declarative category, one based on resolution and the other on reduction.
3. Logic and Functional languages are associated with these two important classes of declarative languages.

Course Outcome:

1. Students will be able to understand Artificial Neural Network concept with the help of Biological Neural Network
2. Students will be able to implement algorithms to train ANN by using learning algorithms
3. Students will be able to test fuzzy set operations and binary relations

UNIT I Introduction of logic & functional paradigm: propositional logic, propositional concepts, natural deduction & axiomatic system, semantic tableaux & resolution FOPL: predicate calculus, Prenex normal forms & skolemization, Herbrand universe & H- interpretation.

UNIT II Logic programming: Execution of query in logic program; programming in PROLOG (overview) meta level & non deterministic programming, second order program in prolog, logic grammars.

UNIT III Functional programming : Multi argument function, expression composition & Equality, higher order functions, curried functions; Introduction to SMIL a Functional language, List, tree, graph manipulation in SML, Recursive data type declaration.

UNIT IV Lambda calculus : Pure Lambda calculus, Currying of function, λ - conversion rules, Applied Lambda calculus, Function definition using λ - notation.

UNIT V Lazy evaluation: Programming with lazy evaluation & delay of unnecessary Computation, infinite data structure, eager evaluation & reasoning, Interactive Functional program.

Text Books:

1. Logic & Prolog programming, Saroj Kaushik, LPE
2. Element of functional Programming, Reade Chris, AWL.

References books :

1. The essence of logic, K. John, PHI

Chhattisgarh Swami Vivekanand Technical University, Bhilai

Name of the programme: **Master of Computer Applications (MCA)**

Subject: **Network Programming (Elective – I)**

Total Theory Periods: **40**

Class Test: **Two (Minimum)**

Duration of ESE: **Three Hours**

Semester: **IV**

Code: **521473 (21)**

Total Tutorial **10**

Periods:

Assignments: **Two (Minimum)**

Minimum Marks in ESE: **40**

Maximum Marks in ESE: **100**

Course Objective:

1. Understand the Network programming concepts like: IPCs, RPC and RMI.
2. Understand the working of RFC based Application Protocols.
3. Understand the real time network based applications at System level.

Ex: Packet Filters.

Course Outcome:

1. Students will be able to conceptualize any given network related problem.
2. Students will be able to understand the technical details given in RFCs for any application

UNIT I Communication protocols: internet protocols, system network architecture, UUCP, XNS, IPX/SPX for LANs, protocol comparisons, TCP & IP headers, IPv4 & v6 address structures.

UNIT II Berkeley Sockets: Overview, socket address structures, value result arguments, byte ordering, byte manipulation & address conversion functions, elementary socket system calls – socket, connect, bind, listen, accept, fork, exec, close, TCP ports (ephemeral, reserved)

UNIT III Berkeley Sockets: I/O asynchronous & multiplexing models, select & poll functions, signal & fcntl functions, socket implementation (client & server programs), UNIX domain protocols, passing the descriptors, IPC: - PIPES, FIFOS, STREAMS & MESSAGES, Semaphore, Shared Memory.

UNIT IV Winsock Programming: windows socket API, window socket & blocking I/O model, blocking sockets, blocking functions, timeouts for blocking I/O, DLL & new API's, DLL issues.

UNIT V Programming Applications: Time & date routines: Internet Time & Date Client, Network Time Synchronization, Trivial file transfer protocol: Data Formats UDP Implementation, TCP Implementation, Remote login, Ping.

Text Books:

1. Window Socket Programming by Bobb Quinn and Dave Schutes
2. Steven.W.R: UNIX Network Programming, PHI (VOL I& II)

Reference Books:

1. Davis.R.: Windows Network Programming, Addison Wesley
2. NETWORK PROGRAMMING With Windows Socket By Baner .P., PH New Jersey

Chhattisgarh Swami Vivekanand Technical University, Bhilai

Name of the programme:	Master of Computer Applications (MCA)	Semester:	IV
Subject:	Modelling & Simulation (Elective – I)	Code:	521474 (21)
Total Theory Periods:	40	Total Tutorial Periods:	10
Class Test:	Two (Minimum)	Assignments:	Two (Minimum)
Duration of ESE:	Three Hours	Maximum Marks in ESE:	100
		Minimum Marks in ESE:	40

Course Objective:

1. Explain the benefits of simulation and modeling in a range of important application areas.
2. Demonstrate the ability to apply the techniques of modeling and simulation to a range of problem areas.
3. Evaluate a simulation, highlighting the benefits and the drawbacks.

Course Outcome:

1. Students will be able to understand the concepts of modeling and simulation of dynamic systems using variety of formalisms.
2. Students will be able to understand when simulation is important tool and when not.
3. Students will be able to verify and validate various simulation models.
4. Students will be able to know the various Techniques for generating random numbers.
5. Students will be able to optimize a system via Simulation.

- UNIT I System Models:** Concept of a system, system environment, stochastic activities, continuous discrete system modeling, system modeling, type of models, static and dynamic systems, principles used in modeling, system studies.
- UNIT II System Simulation:** Techniques of simulation, monte carlo method, comparison of simulation and analytical methods, numerical computation techniques for continuous and discrete models, distributed leg models, cobweb models, simulation study.
- UNIT III Continuous system simulation:** Continuous system models, differential equation, analog computer, analog methods, digital analog simulation, CSSLS, CSMPIII Language. System dynamics: Historical background exponential, Growth and decay models, modified exponential growth models, logistic curves and generalization of growth models, system dynamics diagrams, dynamo language.
- UNIT IV Discrete system simulation:** Discrete events, representation of time, generation of arrival patterns, simulation of telephone system, delayed calls, simulation programming tasks, gathering statistics, and discrete simulation language.
- UNIT V Simulation Language:** Classification of simulation languages, Introduction to GPSC, general description, action times, choice of paths, simulation of a manufacturing shop, facilities and storage, program control statements, priorities and parameters, numerical attributes, functions, simulation of a supermarket transfer models, GPSS model applied to any application, simulation programming techniques like entry types.

Text Books:

1. W.A. Spriet – Computer Oriented Modeling and Simulation
2. G. Gordan-Systems Simulations, PHI, 1995

Reference Books: -

1. T.A. Payer – Introduction to Simulation
2. Barnes – Modeling and Performance Measurement of computer systems.
3. V. Rajaraman “ Analog Simulation” PHI
4. Banks & Carson

Chhattisgarh Swami Vivekanand Technical University, Bhilai

Name of the programme: **Master of Computer Applications (MCA)**

Semester: **IV**

Subject: **Design & Analysis of Algorithm
(Elective – I)**

Code: **521475 (21)**

Total Theory Periods: **40**

Total Tutorial **10**

Periods:

Class Test: **Two (Minimum)**

Assignments: **Two (Minimum)**

Duration of ESE: **Three Hours**

Maximum Marks in ESE: **100**

Minimum Marks in ESE: **40**

Course objectives:

1. Use the potential method to provide an amortized analysis of previously unseen data structure, given the potential function.
2. Explain why competitive analysis is an appropriate measure for online algorithms.
3. Explain the use of randomization in the design of an algorithm for a problem where a deterministic algorithm is unknown or much more difficult.
4. Design and implement a dynamic programming solution to a problem.
5. Familiarizing students with specific algorithms for a number of important computational problems like sorting, searching and graph etc.
6. To introduce the concept of NP-Complete problems and different techniques to deal with them

Course Outcomes:

1. Students will be able to analyze and compare complexity for different types of algorithms for different types of problems.
2. Student will be able to analyze asymptotic runtime complexity of algorithm including formulating recurrence relation.
3. Students will be able to visualize and interpret the domain of problem
4. Students will be able to compare and Validate by adopting different algorithms using more complex data structures for heterogeneous applications.
5. Student will be able to apply algorithm design techniques to solve certain NP-Complete problems.

- UNIT I** **Introduction & Analysis:** Analyzing algorithms, Growth function : Asymptotic notation, Standard notation & common functions, Recurrence relation heaps and introduction to 2-3 trees, Algorithms for manipulating 2-3 trees, Representation of heaps using 2-3 trees, Heap sort , Amortized Analysis.
- UNIT II** **Dynamic Programming Paradigm:** The basic dynamic programming paradigm, Viewing shortest path algorithms from that perspective, Dynamic programming solution to the optimal matrix chain multiplication and the longest common subsequence problems, Top down recursive algorithms using tables of solutions of sub problems as an alternative to bottom up general dynamic programming. Greedy Paradigm :The basic greedy strategy & computing minimum spanning trees, Algorithms of Kruskal and Prim, Use of Union Find Algorithm in implementation of Kruskal's algorithms, The relationship in Dijkstra's and Prim's algorithms, Use of greedy strategy in algorithms for the Knapsack problem and Huffman trees.
- UNIT III** **Divide and Conquer Paradigm:** Divide and Conquer recurrence equations and their solutions, Quick and merge sorting techniques from the perspective of their fitting into the divide and conquer paradigm, Linear time selection algorithm, The basic divide and conquer algorithm for matrix multiplication. Basic Graph Algorithms: Representational issues in graphs, Depth first search on graphs, Computation of biconnected components and strongly connected components using the depth first-search paradigm, Topological sorting of nodes of an acyclic graph. Shortest Path Algorithms on Graphs: Bellman- Ford shortest path problem, Dijkstra's algorithm & Analysis of Dijkstra's algorithm using Fibonacci heaps, Floyd-Warshall's all pairs shortest path algorithm and its refinement for computing the transitive closure of a graph.
- UNIT IV** **String Matching Algorithms:** Modelling the general string problem as a finite automata, Motivation of the failure function in the Knuth Morris and Pratt Paradigm, Linear time analysis of the KMP algorithm, The Boyer-Moore refinement of the KMP algorithm, computation of the failure functions for the Boyer-Moore algorithm.
- UNIT V** **NP-Complete Problems:** Examples of problems like traveling salesman tour for which enumeration and back tracking seems to be the only method of finding the optimal solution, The notion of a non deterministic algorithm and its basic relationship to back tracking. The notion of a polynomial time nondeterministic algorithm, Polynomial time non deterministic algorithms for problems like satisfiability, clique problem, Hamiltonian path problems etc. , The definition of Np-hardness and Np-completeness, The statement of Cook's theorem and a discussion of its implications, The notion of polynomial transformation and reductions, Reductions to show that the clique problem, vertex cover, subset sum and Hamiltonian cycle problems are NP-complete.

Text Books:

1. Introduction to Algorithms (Second Edition) PHI, Corman, Lelerson, Rivertand Stein
2. Fundamentals of Algorithms, Galgotia, Sahni& Horowitz.
3. Analysis of Algorithms, Pearson Education, Hoffcroff.

Reference Books:

Fundamentals of Computer Algorithms, Ellis Horowitz and SatarjShani, Computer Science Press Introduction to Algorithms: A Creative Approach, UdiManberAddision Wesley.

Name of the Programme: Master of Computer Applications (MCA) ::: Duration of the Programme: Three years

Chhattisgarh Swami Vivekanand Technical University, Bilai

Name of the programme: **Master of Computer Applications (MCA)**

Subject: **System programming (Elective – I)**

Total Theory Periods: **40**

Class Test: **Two (Minimum)**

Duration of ESE: **Three Hours**

Semester: **IV**

Code: **521476 (21)**

Total Tutorial **10**

Periods:

Assignments: **Two (Minimum)**

Minimum Marks in ESE: **40**

Maximum Marks in ESE: **100**

Course Objective:

1. To make students understand concept of device drivers.
2. To make students understand debugging and memory management.
3. To make students understand the concept of interrupt handling and race condition.

Course Outcomes:

1. Students will be able to design the drivers.
2. Students will be familiar with architecture of memory models.
3. Students will gain knowledge of interrupt handling and data types in kernel

UNIT I Introduction to Device Drivers: Role of Device Drivers, splitting the kernel, classes of Devices and modules, security issues, version numbering, building and running modules Kernel modules Vs. Applications, compiling & loading, kernel symbol table, preliminaries, Interaction and shutdown, module parameters, doing it in user space.

Char Drivers: Design of scull, major and minor numbers, important data structures, char device registration, open and release, memory usage, read, write, playing with the new devices.

UNIT II Introduction to the Kernel: Important Data structures, Main Algorithms, Implementation of system calls. **Debugging Techniques:** Design support in the Kernel, debugging by printing, querying, watching, system faults. **Memory Management:** Architecture Independent memory models, Virtual address space of a process, block device caching, Paging under Linux. Allocating memory – Kmalloc, look aside caches, get free page and friends, vmalloc and friends, per – CPU variables, obtaining large Buffers.

UNIT III Concurrency and race Conditions: Pitfalls in scull, concurrency & its management, semaphores and mutexes, completions, spin locks, loading traps, alternatives to Locking. Time, Delays, Deferred Work: Measuring time lapses, Knowing current time, delaying execution, kernel timers, tasklets, workqueues.

UNIT IV Interrupt handling: Preparing the parallel port, installing an Interrupt handler, implementing a handler, Top and bottom Halves, Interrupt Sharing, interrupt driven I/O. Communicating with H/W: I/O ports and I/O Memory, Using I/O ports, An I/O port example, using I/O memory.

UNIT V Data types in Kernel: Uses of structured C types, assigning an explicit size to data items, interface specific types, other portability issues, linked lists. Advanced Cache Driver Operations: ioctl, blocking I/O, poll and select, Asynchronous Notification, Seeking a Device, Access control on a device file.

Text Books:

1. Linux Device Drivers- 3rd Edition, J. Corbet, Rubini& Greg K. Hartman, O'Reilly.
2. Linux Kernel Programming – Third Edition, M. Beck et al, Pearson Education.

Reference Book:

Running Linux, 4th edition, Welsh et al, O'Reilly

Name of the Programme: Master of Computer Applications (MCA) ::: Duration of the Programme: Three years

Chhattisgarh Swami Vivekanand Technical University, Bhilai

Name of the programme: **Master of Computer Applications (MCA)**

Semester: **IV**

Subject: **Distributed Operating Systems
(Elective – I)**

Code: **521477 (21)**

Total Theory Periods: **40**

Total Tutorial **10**

Periods:

Class Test: **Two (Minimum)**

Assignments: **Two (Minimum)**

Duration of ESE: **Three Hours**

Maximum Marks in ESE: **100**

Minimum Marks in ESE: **40**

Course objectives:

1. This course provides an introduction to the fundamentals of distributed computer systems, assuming the availability of facilities for data transmission.
2. The structure of distributed systems using multiple levels of software is emphasized. Specific topics include:
 - distributed algorithms
 - distributed file systems
 - distributed databases,
 - security and protection
 - distributed services such as the world-wide web.

Course Outcomes:

1. Student will be familiar key aspects of distributed file systems.
2. Discuss networked, client-server, distributed operating systems and how they differ from single user operating systems.
3. Student will Identify potential threats to operating systems and the security features design to guard against them.
4. understand and analyse theory and implementation of: processes, resource control (concurrency etc.), physical and virtual memory, scheduling, I/O and files.

UNIT I Fundamentals– Fundamentals of OS, What is Distributed System? Evolution of Distributed Computing System, Distributed Computing System Models , Distributed Computing gaining popularity Issues in Designing Distributed System, Introduction to Distributed Computing Environment, Protocols for Distributed System, Network, Interprocess Communication, Issues in Interprocess Communication

UNIT II Architecture of Distributed System – Introduction, Motivations, Concepts of Distributed System, Process Synchronization, System architecture types, Distributed operating system, NOS, Middleware Communication Networks, Communication primitives, Architectural models of Distributed System.
Synchronization :Introduction, Inherent Limitations of a Distributed System, Lamport's logical clock, Vector clock, Global states, Concept of Process, Process Migration, Threads Clock synchronization, Event ordering, Mutual Exclusion, Deadlock, Election Algorithms Issues in Designing Distributed System and role of middleware in Distributed System.

UNIT III Distributed Mutual Exclusion – Introduction, Classification of mutual exclusion algorithms, Preliminaries, A simple solution to distributed mutual exclusion, non token based algorithms, RicartAgrawala algorithm, Token based algorithms, Suzuki Kasami's broadcast algorithms
Distributed Deadlock detection – Introduction, Preliminaries , Deadlock handling strategies , Issues in deadlock detection and resolution, Control organizations for distributed deadlock detection, Centralized deadlock detection algorithms, Distributed deadlock detection algorithms, Avoidance and Prevention algorithms.

UNIT IV Distributed Resource Management, Concepts of File System, Scheduling Algorithms Distributed File System Introduction , Architecture, Mechanisms for building distributed file system, Design issues, Case studies, Log structured file systems, Google FS **Distributed Shared Memory** – Introduction, Architecture and Motivation, algorithms for implementing DSM, Memory Coherence, Coherence protocols, Design issues, Case studies-Linda. **Distributed Scheduling** – Introduction, Motivation, Issues in load distribution, Components of load distributing algorithms, Stability, Load distributing algorithms, Performance Comparison, Selecting a suitable load sharing algorithms, Requirements for load distributing, Load sharing policies, Task migration.

UNIT V Recovery and Security mechanism, Concepts of Database system Security, Basic concepts of Recovery and Types of Failures Recovery – Introduction, Basic concepts, Classification of failures, Backward and forward error recovery, Backward error recovery, Recovery in concurrent systems, Consistent set of checkpoints, Synchronous and Asynchronous check pointing and recovery. **Fault tolerance** – Introduction, Issues, Atomic actions and committing, Commit Protocols, Non blocking Commit protocols, Voting protocols, Majority based Dynamic Voting protocol, Dynamic vote, Reassignment protocols, Failure Resilient Processes, Reliable communication, Case studies-BAR Fault Tolerance, Targon/32 UNIX

Text books:

1. Distributed O.S Concepts and Design, P.K.Sinha, PHI
2. Advanced concepts in Operating Systems , MukeshSinghal&N.G.Shivaratri, TMH
3. Distributed Computing , SunitaMahajan, Seema Shah, OXFORD University Press

Reference Books:

1. Distributed System Principles and Paradigms , Andrew S. Tanenbaum, 2nd edition , PHI
2. Distributed Systems , Colours , 3rd Edition

Name of the Programme: Master of Computer Applications (MCA) ::: Duration of the Programme: Three years

Chhattisgarh Swami Vivekanand Technical University, Bhilai

Name of the programme:	Master of Computer Applications (MCA)	Semester:	IV
Subject:	Cloud Computing (Elective – I)	Code:	521478 (21)
Total Theory Periods:	40	Total Tutorial Periods:	10
Class Test:	Two (Minimum)	Assignments:	Two (Minimum)
Duration of ESE:	Three Hours	Maximum Marks in ESE:	100
		Minimum Marks in ESE:	40

Course Objective:

1. Cloud computing represents a latest in the long history computing mainframe, Personal computing networked computing and expected to revolutionize the business is done.
2. This course covers the theoretical and practical aspects of cloud computing. At the end of the course, student will be able to appreciate the cloud computing paradigm, recognize its various forms and able to implement some cloud computing features.

Course Outcome:

1. Students will be able to perform cloud oriented analysis.
2. Students will be able to model cloud candidate derived from existing business documentation.
3. Students will be able to design the composition of a cloud services.
4. Students will be able to design application services for technology abstraction.

UNIT I Introduction to Cloud Computing, The Emergence of Cloud Computing, Cloud-Based Service Offerings, Benefits of using a Cloud Model, Key Characteristics of Cloud Computing, Understanding-Public & Private cloud environments, The Evolution of Cloud Computing – Hardware & Internet Software Evolution.

UNIT II Cloud Security Challenges, Software-as-a-Service, Security Management People, Security Governance, Security Portfolio Management, Security Architecture Design, Identity Access Management (IAM), Data Security.

UNIT III Cloud as: Communication-as-a-Service (CAAS), Infrastructure-as-a-Service (IAAS), Monitoring-as-a-Service (MAAS), Platform-as-a-Service (PAAS), Software-as-a-Service (SAAS).

UNIT IV The MSP Model, Evolution from the MSP Model to Cloud Computing and Software-as-a-Service, The Cloud Data Center, Basic Approach to a Data Center-Based SOA, Open Source Software, Service-Oriented Architectures as a Step Toward Cloud Computing.

UNIT V Virtualization concepts&Smartphone: virtualization benefits, Hardware virtualization, Software Virtualization, Memory Virtualization, Storage Virtualization, Data Virtualization, Network Virtualization, Virtualization Security Recommendations, Introduction to Various Virtualization OS VMware, KVM, Virtual Machine Security, Smartphone, Mobile Operating Systems for Smartphone's (iPhone, Windows Mobile), Google(Android) Blackberry, Ubuntu Mobile Internet.

Text Books:

1. Toby Velte, Anthony Vote and Robert Elsenpeter, "Cloud Computing: A Practical Approach", McGraw Hill, 2002.

Reference Books:

1. George Reese, "Cloud Application Architectures: Building Applications and Infrastructures in the Cloud", O'Reilly Media, 2003.
2. Tim Matherm, SubraKumaraswamy and ShahedLatif, "Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance", O'Reilly Media, 2005.

Name of the Programme: Master of Computer Applications (MCA) ::: Duration of the Programme: Three years

Chhattisgarh Swami Vivekanand Technical University, Bhilai

Name of the programme: **Master of Computer Applications (MCA)**

Subject: **Biometric Security (Elective – I)**

Total Theory Periods: **40**

Class Test: **Two (Minimum)**

Duration of ESE: **Three Hours**

Semester: **IV**

Code: **521479 (21)**

Total Tutorial **10**

Periods:

Assignments: **Two (Minimum)**

Minimum Marks in ESE: **40**

Course Objective:

1. An understanding of the principles used in biometrics algorithms and systems a knowledge of the most important biometric approaches.
2. The capability to select a suitable algorithm / system for a given application context (e.g. physical access control)
3. A good understanding of the complex relationships between biometric systems and environmental conditions (e.g. illumination, pose variations etc.) and their impact on biometric performance.
4. The capability to assess the security properties of a biometric system. An understanding of data privacy principles and the impact on the design and configuration of biometric systems.

Course Outcome:

1. Students will be able to understand biometric terms
2. Students will be able to gain a broader knowledge of biometric technologies
3. Students will be able to understand differences among biometric technologies
4. Students will be able to design better biometric systems

UNIT I INTRODUCTION OF BIOMETRICS: Benefits of Biometrics over traditional authentication systems and identification systems, Difference between Identification and Verification, Selecting a Biometric for a system, Biometric Applications, Key Biometric terms and processes, Matching process of Biometrics, Limitations and Accuracy measures in Biometric systems – False match rate – False non-match rate – Failure to enroll rate – Derived metrics – Layered biometric solutions.

UNIT II PHYSIOLOGICAL BIOMETRIC TECHNOLOGIES: Fingerprints: Technical description, Characteristics, Strengths, Weaknesses, Deployment, Facial scan: Technical description, Characteristics, Strengths, Weaknesses, Deployment, Iris Scan: Technical description, Characteristics, Strengths, Weaknesses, Deployment, Retina Vascular pattern: Technical description, Characteristics, Strengths, Weaknesses, Deployment, Hand scan: Technical description, Characteristics, Strengths, Weaknesses, Deployment.

UNIT III BEHAVIORAL BIOMETRIC TECHNOLOGIES: Handprint Biometrics, DNA Biometrics, Signature and Handwriting technology: Technical description, Classification, Keyboard / Keystroke dynamics, Voice: Data Acquisition (VDA), Feature Selection and Extraction, Characteristics, Strengths, Weaknesses, Deployment.

UNIT IV MULTI BIOMETRICS, Multi Biometrics and multi factor Biometrics, Two-factor Authentication with passwords, Tickets and Tokens, Executive decision, Implementation plan.

UNIT V BIOMETRICS APPLICATION, Biometric Solution Matrix, Bio privacy – Comparison of privacy factor in different biometrics technologies, Biometrics for Network Security.

Text / Reference Books :

1. Samir Nanavathi, Michel Thieme, and Raj Nanavathi, “Biometrics - Identity Verification in a Network”, Wiley Eastern, 2002.
2. Paul Reid Biometrics for Network Security-, Pearson Education.
3. John Berger, “Biometrics for Network Security”, Prentice Hall, 2004.

Name of the Programme: Master of Computer Applications (MCA) ::: Duration of the Programme: Three years