GURU JAMBHESHWAR UNIVERSITY OF SCIENCE & TECHNOLOGY, HISAR
(Established by State Legislature Act 17 of 1995)
'A' Grade, NAAC Accredited State Govt. University

Dy. R. C&E Gate
Diary No. 4786
Dated 9/9/23

Acad./AC-III/Fac.-1/2023/ 5152
Dated: 6/9/2023

To

The Controller of Examinations
GJUS&T, Hisar.

Sub: Approval of scheme of examinations and syllabi of M.Sc. (Computer Science) programme – 2nd year w.e.f. academic session 2022-23 being run in affiliated degree Colleges.

Sir,

I am directed to inform you that the Vice-Chancellor, on the recommendations of Dean, Faculty of Engg. & Technology, is pleased to approve the scheme of examinations and syllabi of M.Sc. (Computer Science) programme – 2nd year w.e.f. academic session 2022-23 being run in affiliated degree Colleges, under Section 11(5) of the University Act, 1995 in anticipation of approval of the Academic Council.

A copy of the scheme of examinations & syllabi of above said programme is enclosed herewith.

You are therefore, requested to take further necessary action accordingly.

Yours faithfully

Assistant Registrar (Academic)
for Registrar

DA: As above

Endst. No. Acad./AC-III/Fac.-1/2023/ 5153-5156 Dated: 6/9/2023

A copy of the above is forwarded to the following for information and necessary action:-

1. Dean, Faculty of Engg. & Technology, GJUS&T, Hisar.
2. Chairperson, Department of Computer Science & Engg., GJUS&T, Hisar alongwith scheme of examinations and syllabi of M.Sc. (Computer Science) programme – 2nd year w.e.f. academic session 2022-23 being run in affiliated degree Colleges being run in affiliated degree Colleges. He is requested to arrange to upload the scheme of examinations & syllabi of above said programme on the website of the University immediately.
3. Principal, Government College, Hisar alongwith M.Sc. (Computer Science) programme – 2nd year w.e.f. academic session 2022-23 being run in affiliated degree Colleges.
4. Principal, ODM College for Women, Hisar alongwith M.Sc. (Computer Science) programme – 2nd year w.e.f. academic session 2022-23 being run in affiliated degree Colleges.

Assistant Registrar (Academic)

Scheme and Syllabus

of

~~Master of Computer Science~~

M.Sc. (Computer Science)

2 YEARS PROGRAMME

CHOICE BASED CREDIT SYSTEM (CBCS)

(70:30)

(w.e.f. Academic Session 2022-23)

Guru Jambheshwar University of Science & Technology, Hisar

Scheme of Examination for ~~Master of Computer Science~~

M.Sc. (Computer Science)
Two-Year Programme under CBCS Scheme
w.e.f. Academic Session 2022-2023

M.Sc. SEMESTER-I

Course Code	Course Title	Credit	Int.	Ext.	Total
MCS-11	Database Management System	3	30	70	100
MCS-12	OOPS using C++	3	30	70	100
MCS-13	Discrete Mathematics	3	30	70	100
MCS-14	Computer Organization and Architecture	3	30	70	100
MCS-15	Computer Networks	3	30	70	100
MCS-16	Database Management System Lab	2	30	70	100
MCS-17	C++ Lab	2	30	70	100
MCS-18	Computer Network Lab	2	30	70	100
Total		**21**	**240**	**560**	**800**

M.Sc. SEMESTER–II

Course Code	Course Title	Credit	Int.	Ext.	Total
MCS -21	Data Structures and Algorithms	3	30	70	100
MCS -22	Java Programming	3	30	70	100
MCS -23	Operating System Concepts	3	30	70	100
MCS -24	Computer Graphics	3	30	70	100
MCS -25	Theory of Computations	3	30	70	100
MCS -26	Data Structures and Algorithm Lab	2	30	70	100
MCS -27	Java Programming Lab	2	30	70	100
MCS -28	Computer Graphics Lab	2	30	70	100
Total		**21**	**240**	**560**	**800**
Four to six weeks Industrial/Summer training at the end of second semester					

M.Sc. SEMESTER-III

Course Code	Course Title	Credit	Int.	Ext.	Total
MCS -31	Artificial Intelligence	3	30	70	100
MCS -32	Python Programming	3	30	70	100
MCS -33	Software Engineering	3	30	70	100
MCS -34	Elective–I	3	30	70	100
MCS -35	Elective–II	3	30	70	100
MCS -36	Artificial Intelligence Lab.	2	30	70	100
MCS -37	Python Programming Lab.	2	30	70	100
MCS -38	Industrial/Summer Training	2	100	---	100
Total		**21**	**310**	**490**	**800**

MCS–34 Elective–I List of Courses

MCS – 34 (i) Management Information System
MCS – 34 (ii) ADBMS
MCS – 34 (iii) Analysis and Design of Algorithms

MCS–35 Elective–II List of Courses

MCS – 35 (i) Multimedia and Applications
MCS – 35 (ii) Advanced Computer Architecture
MCS – 35 (iii) High Speed Networks

M.Sc. SEMESTER-IV

Course Code	Course Title	Credit	Int.	Ext.	Total
MCS -41	IoT and Cloud Computing	3	30	70	100
MCS -42	Data Warehouse and Data Mining	3	30	70	100
MCS -43	Web Development	3	30	70	100
MCS -44	Elective–III	3	30	70	100
MCS -45	Elective–IV	3	30	70	100
MCS -46	IoT and Cloud Computing Lab	2	30	70	100
MCS -47	Project Work	4	30	70	100
Total		**21**	**210**	**490**	**700**

MCS–44 Elective–III List of Courses

- **MCS – 44(i)** Software Project Management
- **MCS – 44(ii)** Big Data Analytics
- **MCS – 44(iii)** Cyber Security

MCS – 45 Elective–IV List of Courses

- **MCS – 45 (i)** Soft Computing
- **MCS – 45 (ii)** Compiler Design
- **MCS – 45 (iii)** Mobile Application Development

Total Credits of Programme

w.e.f. Academic Session 2022-2023 **M.Sc. 2–Years under CBCS**

Semester	Max. Marks	Credits
I	800	21
II	800	21
III	800	21
IV	700	21
Programme Total	**3100**	**84**

Note:

1. Evaluation of Industrial/Summer Training is done by two members' internal committee constituted by Head of the Department.
2. Evaluation of Project Work is done by the internal examiner (project guide) and external examiner appointed by Controller of Examination.

MCS-31 Artificial Intelligence

General Course Information

Course Code: MCS-31 Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Three minor examinations (20 marks each) including third minor in open book mode will be conducted. The average of the highest marks obtained by a student in the any of the two minor examinations will be considered. Class Performance measurement through percentage of lectures attended (4 marks), assignments (6 marks), and the end-semester examination (70 marks) will be considered. For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basic knowledge of Algorithms and probability.

About the Course: Artificial Intelligence is a core and an essential course for every graduate in Computer Science and Engineering. This course introduces the concepts of Artificial Intelligence and challenges inherent in building intelligent systems. It includes the role of knowledge representation in problem solving and how these are used in making intelligent machine. Further it incorporates the concepts of expert system and its applications.

Course Outcomes: By the end of the course students will be able to:

CO1. **study** the various Artificial Intelligence techniques. (LOTS: Level 1: Remember)
CO2. **illustrate** reasoning under uncertainty. (LOTS: Level 2: Understand)
CO3. **apply** search and knowledge representation techniques to solve AI problems.(LOTS: Level 3: Apply)
CO4. **compare** strengths and weaknesses of AI algorithms (HOTS: Level 4: Analyse).
CO5. **combine** various AI techniques to solve intelligent systems' problems. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction to AI: Introduction, Turing Test, AI problems, State Space Search, production system Problem Solving Using Search: Blind search techniques - Breadth first search, Depth first search. Heuristic search techniques - Generate and test, Hill Climbing, Best first search, A* Algorithm, AO* Algorithm, The Minimax Search Procedure, Adding Alpha-Beta Cut-offs.

Unit II

Knowledge Representation: Introduction, Knowledge Representation- Representation and Mappings, Symbolic Logic - Propositional logic, Predicate logic- Representing simple facts in logic, Representing Instances and ISA Relationship, Computable functions and Predicates, Unification, Resolution. 143 Representing Knowledge Using Rules: Procedural versus Declarative Knowledge, Logic Programming, Forward versus Backward Reasoning, Matching, Control Knowledge.

Unit III
Reasoning Under Uncertainty: Introduction to Nonmonotonic Reasoning, Probability and Baye's Theorem, Certainty Factors and Rule-based Systems, Bayesian Networks. Fuzzy logic system: Introduction, Crisp Set, Fuzzy Sets, Fuzzy Membership Functions, Operations on Fuzzy Sets, Fuzzy Relations.

Unit IV
Planning: Introduction, Components of Planning System, Goal Stack Planning, Nonlinear Planning using Constraint Posting, Hierarchical Planning. Expert System and Applications: Introduction, Architecture, Rule based Expert Systems, Applications of Expert Systems.

Text and Reference Books:
1. Elaine Rich, Kevin Knight and Shivashankar B Nair, Artificial intelligence, McGraw Hill Education. 3 rd edition, 2009.
2. Stuart Russel and Peter Norvig, Artificial intelligence: A modern Approach, Pearson Education, 3rd edition, 2015.
3. Dan W. Patterson, Introduction to Artificial Intelligence and Expert System, Pearson Education.1st edition, 2007.
4. Deepak Khemani, A first course in Artificial Intelligence, McGraw Hill Education. 3rd edition, 1st edition, 2013.
5. George F. Luger, Artificial Intelligence: Structures and Strategies for Complex Problem Solving, Pearson Education, 5th edition, 2009.

MCS-32 Python Programming

General Course Information

Course Code: MCS-32 Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Three minor examinations (20 marks each) including third minor in open book mode will be conducted. The average of the highest marks obtained by a student in the any of the two minor examinations will be considered. Class Performance measurement through percentage of lectures attended (4 marks), assignments (6 marks), and the end-semester examination (70 marks) will be considered. For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisite: Exposure to programming languages.

About the Course: Python is a popular open source programming language used for both standalone programs and scripting applications in a wide variety of domains. It is free, portable, and powerful and is both relatively easy and remarkably fun to use. In today's era Python has found great applicability in machine learning, data analytics and many other data science application. This is introductory course and covers most of the basic concepts required for basic python programming. Some of the contents are advanced may be useful for data analytics purpose.

Course Outcomes: By the end of the course students will be able to:

CO1. **study** the various basic programming constructs including operators, character sets, basic data types and control statements. (LOTS: level 1: Understand)
CO2. **explain** Python packages and their functionalities for data analysis. (LOTS: level 2: Understand)
CO3. **solve** problems using python programming. (LOTS: level 3: Apply)
CO4. **analyse** the results of data analysis or machine learning programs (HOTS: level 4: Analyse) CO5. **evaluate** solutions according to the problem definition. (HOTS: level 5: Evaluate)
CO6. **develop** database applications in Python. (HOTS: level 6: Create)

Course Content

Unit I

Introduction to Python, History of Python, Features of Python, Python Identifiers, Python Character Set, Keywords and Indentation, Comments, Command Line Arguments, Assignment Operator, Operators and Expressions, print() Function, input() Function, eval() Function, Python Data Types: int, float, complex, Variables, Mutable vs Immutable variables, Namespaces, Decision Statements: Boolean

Type, Boolean Operators, if statement, else statement, Nested Conditionals Statements, Multi-way Decision Statements (elif statement).

Unit II

Loop Control Statements: While loop, range() Function, For Loop, Nested Loops, Infinite Loop, Break Statement, Continue Statement, Pass Statement, Introduction to Strings, String Operations: Indexing and Slicing, Lists: Operations on List: Slicing, Inbuilt Functions for Lists, List Processing: Searching and Sorting, Dictionaries: Need of Dictionary, Operations on Directories: Creation, Addition, Retrieving Values, Deletion; Tuples, operations on Tuples, Inbuilt Functions for Tuples, Introduction to Sets, operations on sets. Python Functions, Inbuilt functions, Main function, User Defined functions, Defining and Calling Function, Parameter Passing, Actual and Formal Parameters, Default Parameters, Global and Local Variables, Recursion, Passing Functions as Data, Lamda Function, Modules, Importing Own Module, Packages.

Unit III

Operations on File: Reading text files, read functions, read(), readline() and readlines(), writing Text Files, write functions, write() and writelines(),Manipulating file pointer using seek, Appending to Files. Python Object Oriented: Overview of OOP, Classes and objects, Accessing attributes, Built-In Class Attributes, Methods, Class and Instance Variables, Destroying Objects, Polymorphism, Overlapping and Overloading of Operators, Class Inheritance: super(), Method Overriding, Exception Handling, Try-except-else clause, Python Standard Exceptions, User-Defined Exceptions

Unit IV

Databases in Python: Create Database Connection, create, insert, read, update and delete Operation, DML and DDL Operation with Databases. Python for Data Analysis: numpy: Creating arrays, Using arrays and Scalars, Indexing Arrays, Array Transposition, Universal Array Function, Array Processing, Array Input and Output Pandas: Series, Data Frame, Panel, Index objects, Re-indexing, Iteration, Sorting. Matplotlib: Python for Data Visualization, Visualization Section, Sklearn: loading of dataset, learning and predicting, Model Persistence.

Text and Reference Books:

1. Ashok Namdev Kamthane, Programming and Problem Solving with Python, Mc Graw Hill Education Publication, 2018.
2. John Guttag, Introduction to Computation and Programming using Python, Springer, Revised and Expanded version (Referred by MIT), 2013.
3. Lutz, M., Learning Python: Powerful Object-Oriented Programming. O'Reilly Media, Inc., 2013.
4. Michael T Goodrich and Robertto. Thamassia, Micheal S Goldwasser, Data Structures and Algorithms in Python, Wiley, 2016.
5. Y. Daniel Liang, Introduction to Programming Using Python, Pearson, 2013.
6. Reema Thareja, Python Programming Using Problem Solving Approach , Oxford Publications, 2017.
7. Dr. R. Nageswara Rao, Allen B. Downey, Core Python Programming , Think Python, O'Reilly Media, 2012.
8. Kenneth A. Lambert, The Fundamentals of Python: First Programs, Cengage Learning, 2011.

MCS-33 Software Engineering

General Course Information

Course Code: MCS-33 Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Three minor examinations (20 marks each) including third minor in open book mode will be conducted. The average of the highest marks obtained by a student in the any of the two minor examinations will be considered. Class Performance measurement through percentage of lectures attended (4 marks), assignments (6 marks), and the end-semester examination (70 marks) will be considered. For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Knowledge of algorithms, flow charts and a programming language.

About the Course: Software Development is generally a quite complex and time-consuming process. Moreover, depending on the nature and complexity of the software requirements, Software Engineering plays an important role. This course will help the students to understand the systematic approach to requirement analysis, design, development, operations and maintenance of software systems. Besides this, it also guides students in developing the optimal software systems.

Course Outcomes: By the end of the course students will be able to:

CO1. **define** the various concepts related to software engineering. (LOTS: Level 1: Remember)
CO2. **demonstrate** the use of stages of various Software Life Cycle Models. (LOTS: Level 2: Understanding)
CO3. **apply** the Software Requirement Analysis and Software Design Process. (LOTS: Level 3: Apply)
CO4. **analyse** the size, cost, complexity, reliability, quality and maintenance of a software system. (HOTS: Level 4: Analyse)
CO5. **construct** software model according to the requirements of a customer. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction: Software Crisis, Software Process, Evolution of Software Engineering, Software Characteristics, Software Metrics and SDLC. Software Life Cycle Models: Water Fall Model, Increment Process Model, Evolutionary Process Models, Unified Process. Selection of Life Cycle Model. Software Requirements, Analysis and Specifications: Requirement Engineering, Requirements Elicitation, Requirements Analysis: Data Flow Diagram, Data Dictionary, Entity-Relationship Diagrams, Decision Table, Decision Tree and Structured Charts. Requirements Documentation and Requirements validation.

Unit II

Software Project Management: Size Estimation, Cost Estimation, Constructive Cost Model (COCOMO), Putnam Resource Allocation Model. Software Risk Management: Software Risks, Risk Identification, Risk Projection, Risk Refinement, Risk Mitigation, Monitoring, and Management, RMMM Plan.

Unit III

Software Design: Software Design Fundamentals, Modularity, Design Principles, Strategy of Design, Function Oriented Design, and Object Oriented Design, IEEE Recommended Practice for Software Design Descriptions. Software Quality: Basic Concepts, ISO 9126, McCall's Quality Factors, Software Quality Assurance, SQA Activities, Software Review Process, Formal Technical Review, ISO 9000 Quality Standards, and CMM.

Unit IV

Software Testing: Testing fundamentals, Verification and Validation, Test Plan, Test Case, Levels of Software Testing: Unit Testing, Integration Testing, Top Down and Bottom up Testing Integration Testing, Alpha and Beta Testing, System Testing, White Box Testing and Black Box Testing, Debugging and Software Testing Tools. Maintenance and Reengineering: Software Maintenance, Software Supportability, Reengineering, Business Process Reengineering, Software Reengineering, Reverse Engineering, Restructuring, Forward Engineering and The Economics of Reengineering.

Text and Reference Books:

1. K. K. Aggarwal and Yogesh Singh, Software Engineering, 3rd Edition, New Age International Publishers Ltd., Reprint 2014.
2. Roger S. Pressman, Software Engineering: A Practitioners Approach 7 th Edition, Mc Graw Hill Education, 2014.
3. Rajib Mall, Fundamental of Software Engineering, Prentice Hall India, 2004.
4. Pankaj Jalote, An integrated Approach to Software Engineering, 3rd Edition, Narosa Publications, 2014.
5. Ian Sommerville, Software Engineering, 10th Edition, Addison-Wesley, 2015.
6. Carlo Ghezzi, Mehdi Jazayeri and Dino Mandrioli, Fundamentals of Software Engineering, 2nd Edition, Pearson, 2007.
7. Waman S Jawadekar, Software Engineering-Principles and Practice, Tata McGraw-Hill, 2004.

MCS-34 (i) Management Information System
(Elective Course - I)

General Course Information

Course Code: MCS-34 (i) Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Three minor examinations (20 marks each) including third minor in open book mode will be conducted. The average of the highest marks obtained by a student in the any of the two minor examinations will be considered. Class Performance measurement through percentage of lectures attended (4 marks), assignments (6 marks), and the end-semester examination (70 marks) will be considered. For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Knowledge of computer terminologies and programming language(s) will help in understanding the given concepts very easily.

About the course: An effective and efficient MIS is always useful for reporting to the management for timely decisions. This course will help the students to learn about the information needs of an organization, various tools and methods with other needs such as security and control issues to design and develop an effective and efficient MIS for organization(s).

Course Outcomes: By the end of the course students will be able to:

CO1. Study of people, technology, organizations, and the relationships among them.

CO2. Learn various tools and methods required to develop an effective and efficient MIS
CO3. Understand various control and security issues to be taken care at the time of development of an MIS.
CO4. Identify and analyze the information needs of an organization required to design and develop an efficient and effective MIS.
CO5. Design and develop an efficient and effective MIS for an organization.

Course Contents

Unit I

Background Meaning, Nature, Need, Role, Importance, Evolution of management through information system; Relatedness of MIS with management process. Management functions and decision making. Concept of balance MIS effectiveness and efficiency criteria.

Unit II

Development of Management Information System: Introduction, Information system planning, Motivational forces behind development of information system, Principles for information system development, SDLC for MIS development process.

Unit III

Development of MIS: Methodology and Tools techniques for systematic identification, implementation, evaluation, and maintenance of MIS. Control and Security Issues in Management Information Systems: Control, Why need to Control MIS, Types of Control, Audit in MIS, Security Hazards, Security Techniques.

Unit IV

Case studies: To introduce business problems and to discuss various stages for understanding the systems development process.

Text and Reference Books:

1 James A.O'Brien, Management Information Systems.
2 Kenneth C. Laudon, Jane P. Laudon, Ahmed Elragal, Management Information Systems: Managing The Digital Firm, Pearson.
3 S. Sadagopan, Management Information Systems, PHI Learning Pvt. Ltd.; Second edition (2014)
4 Avdhesh Gupta, Anurag Malik, Management Information Systems: A Computerized approach for Managerial Aspects; FireWall Media
5 Indrajit Chatterjee, Management Information Systems, PHI Learning Pvt. Ltd.;(2010)
6 Davendranath G. Jha, Computer Concepts And Management Information Systems, PHI Learning Pvt. Ltd.;(2013)

MCS-34 (ii) ADBMS
(Elective Course -I)

General Course Information

Course Code: MCS-34 (ii) Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Three minor examinations (20 marks each) including third minor in open book mode will be conducted. The average of the highest marks obtained by a student in the any of the two minor examinations will be considered. Class Performance measurement through percentage of lectures attended (4 marks), assignments (6 marks), and the end-semester examination (70 marks) will be considered. For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: One should have the basic knowledge of Database System concepts and basic queries of SQL.

Course Outcomes: By the end of the course students will be able to:

CO1. **define** the various concepts of database applications and administration. (LOTS: Level 1: Remember)

CO2. **demonstrate** the use of various transaction processing and recovery methods. (LOTS: Level 2: Understanding)

CO3. **apply** the transaction processing and recovery methods (LOTS: Level 3: Apply)

CO4. **analyse** the size, cost, complexity and reliability of various transaction processing and recovery methods. (HOTS: Level 4: Analyse)

CO5. **construct** be able to design and manage database effectively using advanced queries of Oracle (HOTS: Level 6: Create)

Course Contents

Unit I

Introduction to Database System: Database System Applications, Normalization: 1NF, 2NF, 3NF, BCNF, 4NF and 5NF, SQL Queries, Database Users and Administrators.

Unit II

Transaction Management: Transactions and Its ACID Properties, Serializability: Conflict Serializability, View Serializability, Testing for Serializability, Concurrency Control Techniques: Lock-Based Protocols, Timestamp Based Protocols, Validation Based Protocol.

Unit III

Recovery System: Failure Classification, Storage Structure: Types, Stable Storage Implementation, Data Access, Log Based Recovery, Advanced Recovery Techniques: Logical Undo Logging, Transaction Rollback, Checkpoints, Remote Backup Systems.

Unit IV

Oracle Concepts: Introduction to SQL *PLUS, Referential Integrity, SQL *PLUS Reports, Introduction to PL/SQL, Cursors, Triggers, Procedures, Functions, Packages, Large Objects, Creating Users, Remote Data Access.

Text and Reference Books:

1. Henry F. Korth, Database System Concepts, Fifth Edition, McGraw-Hill, 2006.
2. Navathe, Fundamentals of Database Systems, Fourth Edition, Pearson Education, 2008.
3. P.S. Deshpande, SQL & PL/SQL for Oracle, Black Book, Dreamtech Press, 2006.
4. Juneau J., Oracle PL/SQL Recipes: A Problem Solving Approach, APress, 2010.

MCS-34 (iii) Analysis and Design of Algorithms
(Elective Course -I)

General Course Information

Course Code: MCS-34 (iii) Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Three minor examinations (20 marks each) including third minor in open book mode will be conducted. The average of the highest marks obtained by a student in the any of the two minor examinations will be considered. Class Performance measurement through percentage of lectures attended (4 marks), assignments (6 marks), and the end-semester examination (70 marks) will be considered. For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Knowledge of Data Structure and a Programming Language

About the Course: This Course focus on effective and efficient design of algorithms. In this course various algorithm design techniques and their analysis is to be studied. After studying this course, a student is expected to apply better techniques for solving computational problems efficiently and prove it analytically.

Course Outcomes: By the end of the course students will be able to:

CO1. **state** terminology and concepts algorithmic techniques. (LOTS: Level 1: Remember)

CO2. **discuss** various algorithmic techniques. (LOTS: Level 2: Understand)

CO3. **apply** appropriate algorithmic techniques to solve computational problems. (LOTS: Level 3: Apply)

CO4. **analysing** algorithms for their efficiency by determining their complexity. (HOTS: Level 4: Analyse)

CO5. **compare** the pros and cons of applying the different algorithmic techniques to solve problems. (HOTS: Level 5: Evaluate)

CO6. **formulate** efficient and effective algorithmic solutions for different real- world problems. (HOTS: Level: 6 Create)

Course Content

Unit I

Algorithms, Algorithms as a technology, Insertion sort, Analyzing algorithms, asymptotic notations, Divide and Conquer: General method, binary search, merge sort, quick sort, Strassen"s matrix multiplication algorithms and analysis of algorithms for these problems.

Unit II

Sorting and Data Structures: Heapsort, Hash Tables, Red and Black Trees, Greedy Method: General method, knapsack problem, minimum spanning trees, single source paths and analysis of these problems.

Unit III

Dynamic Programming: General method, matrix chain multiplication, longest common subsequence, optimal binary search trees, Back Tracking: General method, 8 queen's problem, graph colouring, Hamiltonian cycles, Analysis of these problems.

Unit IV

Branch and Bound: Method, O/I knapsack and traveling salesperson problem, NP Completeness: Polynomial time, NP-completeness and reducibility, NP-complete problems.

Text and Reference Books:

1. Thomas H. Cormen, Charles E. Leiserson, Ronalde L. Rivest, Clifford Stein, Introduction to Algorithms, MIT press, 3rd Edition, 2009.
2. Ellis Horowitz, Satraj Sahni, Sanguthevar Rajasekaran, Fundamental of Computer Algorithms, Galgotia publication Pvt. Ltd., 1999.
3. S. Dasgupta, C. Papadimitriou, and U. Vazirani, Algorithms, McGraw-Hill Higher Education, 2006.

MCS-35 (i) Multimedia and Applications
(Elective Course -II)

General Course Information

Course Code: MCS-35 (i) Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Three minor examinations (20 marks each) including third minor in open book mode will be conducted. The average of the highest marks obtained by a student in the any of the two minor examinations will be considered. Class Performance measurement through percentage of lectures attended (4 marks), assignments (6 marks), and the end-semester examination (70 marks) will be considered. For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basics of Computer Graphics

About the Course: Multimedia is a core and an essential course for every graduate in Computer Science and Engineering The objective of this course is to make students learn how to develop multimedia programs and demonstrate how still images, sound, and video can be digitized on the computer.

Course Outcomes: By the end of the course students will be able to:

CO1. **study** the basic concepts of multimedia technology. (LOTS: Level 1: Remember)

CO2. **discuss** the concepts of animation, digitized sound, video control, and scanned images. (LOTS: Level 2: Understand)

CO3. **use** basic instructional design principles in the development of Multimedia. (LOTS: Level 3: Apply)

CO4. **compare** various audio and video file formats. (HOTS: Level 4: Analyse)

CO5. **devise** solutions for multimedia problems. (HOTS: Level 6: Create)

Course Content

Unit 1

Introduction to Multimedia concepts, Types of Multi-media Applications, Methods to deliver Multimedia, Introduction to Multimedia Database, Multimedia Input and Output Devices.

Unit II

Introduction about font and faces, Using Text in Multimedia, Applying different types of text in multimedia Font Editing and Design tools, Hypermedia and Hypertext application. 164

Unit III

The power of images, Making Still Images, Colouring, Image File Formats (GIF, JPEG, PNG etc.) The power of sound, MIDI Vs. Digital Audio, Audio File Formats (AIFF, WAV, MPEG, MOV etc.) Adding Sound to multimedia project.

Unit IV

Working of a Video and its Display, Digital Video Containers (Codecs & Video Format Converters) Obtaining Video Clips, Shooting and editing Video, Non Linear Editing (NLE) in Videos The stages of Multimedia Project, Hardware and Software requirements, Authoring Systems Team for Multimedia Development, Different stages of multimedia, The internet and multimedia.

Text and Reference Books:

1. Tay Vaughan, Multimedia: Making It Work, Tata McGraw Hills, 2008.
2. James E Shuman, Multimedia in Action, Vikas Publishing House, 1997.
3. Andreas Holzinger, Multimedia Basics Technology, Volume 1, Firewall Media, 2005.
4. Rangan Parekh, Principles of Multimedia, Tata McGraw Hills, 2007.

MCS-35 (ii) Advanced Computer Architecture
(Elective Course -II)

General Course Information

Course Code: MCS-35 (ii) Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Three minor examinations (20 marks each) including third minor in open book mode will be conducted. The average of the highest marks obtained by a student in the any of the two minor examinations will be considered. Class Performance measurement through percentage of lectures attended (4 marks), assignments (6 marks), and the end-semester examination (70 marks) will be considered. For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Digital Electronics and computer systems.

About the Course: Computer Architecture and organization describes the role of instruction set architecture in digital computer, main memory, and input/output devices. It illustrates the simple data path and control design for processors. It helps to understand the different operations and concept of instructions. It would enable the students to learn the basic function and architecture of modern computer systems.

Course Outcomes: By the end of the course students will be able to:

CO1. **study** the general concepts of digital electronics and computer organisation and architecture. (LOTS: Level 1: Remember)

CO2. **discuss** the basic components and their interfacing.(LOTS: Level 2: Understand)

CO3. **apply** instructions for performing different operations. (LOTS: Level 3: Apply)

CO4. **analyse** the effect of addressing modes on the execution time of a program.(HOTS: Level 4: Analyse)

CO5. **contrast** different types of memory, their architecture and access methods. (HOTS: Level 5: Evaluate)

CO6. **design** of simple computer with different instruction sets. (HOTS: Level 6: Create)

Course Content

Unit I

Basic Principles: Boolean algebra and Logic gates, Combinational logic blocks (Adders, Subtractors, Multiplexers, Encoders, decoders, demultiplexers, KMaps), Sequential logic blocks (Flip-Flops, Registers, Counters); Flynn's classification of computers (SISD, MISD, MIMD); Performance metrics: MIPS, MFLOPS; CPU Architecture types: computer register, (accumulator, register, stack, memory/ register) detailed data path of a typical register based CPU.

Unit II

Computer Organization: Store program control concept, Instruction codes, timing and control, instruction cycle; type of instructions: memory reference, register reference, I/O reference; Basics of Logic Design, accumulator logic, Control memory; Micro Programmed Control: address sequencing, micro-instruction formats, micro-program sequencer, Implementation of control unit.

Unit III

Instruction Set Architecture & Parallelism: Instruction set based classification of processors (RISC, CISC, and their comparison); Stack Organization, Instruction Formats; addressing modes: register, immediate, direct, indirect, indexed; Operations in the instruction set: Arithmetic and Logical, Data Transfer, Control Flow; Types of interrupts; Introduction to Parallelism: Goals of parallelism (Exploitation of concurrency, throughput enhancement); Amdahl's law; Instruction level parallelism (pipelining, super scaling –basic features); Processor level parallelism (Multiprocessor systems overview).

Unit IV

Memory Hierarchy & I/O Techniques: The need for a memory hierarchy (Locality of reference principle, Memory hierarchy in practice: Cache, main memory and secondary memory, Memory parameters: access/ cycle time, cost per bit); Main memory (Semiconductor RAM & ROM organization, memory expansion, Static & dynamic memory types); Cache memory (Associative & direct mapped cache organizations; input-output interface, mode of transfer, DMA (Direct memory transfer).

Text and Reference Books:

1. Mano, M. Morris, Digital Logic and Computer Design, Prentice Hall of India Pvt. Ltd., 1981.
2. M. Morris Mano, Computer System Architecture, Prentice Hall of India Pvt. Ltd., 1993.
3. Milles J. Murdocca, Vincent P. Heuring, Computer Architecture and Organization, An Integrated Approach, JohnWiley & Sons Inc., 2007.
4. William Stallings, 10th edition, Computer Organization and Architecture, Prentice Hall, 2016.
5. Heuring, V.P., Jordan, H.F., Computer Systems Design and Architecture, Addison Wesley,1997.
6. R.P Jain, Modern Digital Electronics, 3rd Edition , Tata McGraw Hill,, 2003.

MCS-35 (iii) High Speed Networks
(Elective Course -II)

General Course Information

Course Code: MCS-35 (iii) Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Three minor examinations (20 marks each) including third minor in open book mode will be conducted. The average of the highest marks obtained by a student in the any of the two minor examinations will be considered. Class Performance measurement through percentage of lectures attended (4 marks), assignments (6 marks), and the end-semester examination (70 marks) will be considered. For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basic knowledge of computer networks, layers of OSI reference model, protocols at different layers of OSI reference model.

About the course: High Speed Network Technologies is a professional core course based around Network Architectures, protocols used across the layers, techniques used in communication and modes of data transfer. The course deals with creating High Speed Networks for any organization/institute with its various phases/life cycles.

Course Outcomes: By the end of the course students will be able to:
CO1. **define** different high speed network technologies. (LOTS: Level 1: Remember)
CO2. **explain** working of different wired / wireless technologies suitable for LAN and WAN communication. (LOTS: Level 2: Understand)
CO3. **illustrate** the mapping of OSI reference model to different high speed technologies and Internet Suite of Protocols. (LOTS: Level 3: Apply)
CO4. **analyze** the performance of different high speed technologies in different scenarios / situations. (HOTS: Level 4: Analyse)
CO5. **design** a network for any organization using high speed technologies along with Internet connectivity. (HOTS: Level 6: Create)

Course Content

Unit I
(High Speed LAN) Gigabit Ethernet: Overview of fast Ethernet, Gigabit Ethernet – overview, specifications, layered protocol architecture, frame format, network design using Gigabit Ethernet, applications, 10GB Ethernet – overview, layered protocol architecture, frame format. Fiber Channel: Fiber channel – overview, topologies, ports, layered protocol architecture, frame structure, class of service.

Unit II

(High Speed WAN) Frame Relay: Protocol architecture and frame format. ISDN & B-ISDN: Channels, interfaces, addressing, protocol architecture, services. ATM: Virtual circuits, cell switching, reference model, traffic management.

Unit III

(Wireless LAN) Wireless Networks: Existing and emerging standards, Wireless LAN (802.11), Broadband Wireless (802.16), Bluetooth (802.15) their layered protocol architecture and security. Mobile Networks – GSM, CDMA.

Unit IV

(Internet Suite of Protocols) Internet Layer: IPV4 and IPV6, IP addressing, IP classes, CIDR. Transport Layer: UDP/TCP protocols & architecture, TCP connection management. Application Layer: DNS, E-Mail, Voice over IP.

Text and Reference Books:

1. Jochen Schiller, Mobile Communication, 2nd Edition, Pearson, 2009.
2. Andrew S Tanenbaum, Computer Networks, 5th Edition, Pearson 2013.
3. William C Y Lee, Mobile Communication Engineering: Theory and Applications, 2nd Edition, McGraw Hill, 1997.

General Course Information

MCS-36 Artificial Intelligence Lab.

Course Code: MCS-36 Course Credits: 2 Type: Professional Core Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external:70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner (appointed by the Controller of Examination) in association with the internal examiner appointed by the Head of the Department.

Prerequisites: Basic knowledge of HTML, XML, ASP, JSP and Web Designing.

About the Course: In this course, the learners will be able to develop expertise related to general purpose problem solving, Representation of knowledge, Reasoning under uncertainty, Planning and Natural Language processing

Course Outcomes: By the end of the course students will be able to:

CO1. **understand** the various Artificial Intelligence techniques.
CO2. **illustrate** reasoning under uncertainty.
CO3. **apply** search and knowledge representation techniques to solve AI problems.
CO4. **compare** strengths and weaknesses of AI algorithms.
CO5. **combine** various AI techniques to solve intelligent systems' problems.

List of experiments/ assignments:

1. Write a program to implement BFS/DFS Traversal?
2. Write simple facts for the statements and querying it.
3. Write a program for Family-tree.
4. Write Program for Monkey-banana Problem.
5. Write a program to implement Tic-Tac-Toe game.
6. Write programs for computation of recursive functions like factorial Fibonacci numbers, etc.
7. Write program to solve 5-queens problem.
8. Write a Program for water jug problem.
9. Write a program for travelling salesman problem.
10. Write a program to implement all set operations.

Note:
The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

MCS-37 Python Programming Lab.

General Course Information

Course Code: MCS-37 Course Credits: 2 Type: Professional Core Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external:70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner (appointed by the Controller of Examination) in association with the internal examiner appointed by the Head of the Department.

Pre-requisites: Basic programming skills

About the Course: Python is a scripting programming language known for both its simplicity and wide breadth of applications. For this reason it is considered one of the best languages for beginners. Used for everything from web development to scientific computing Python is referred to as a general purpose language by the greater programming community. The major objective of Python language is to make the students solve real word problem efficiently using python library.

Course Outcomes: By the end of the course students will be able to:

CO1. **implement** solutions to the given assignments in Python. (LOTS: Level 3: Apply)
CO2. **use** various Python packages for solving different programming problems. (LOTS: Level 3: Apply)
CO3. **devise** solutions for complex problems of data analysis and machine learning. (HOTS: Level 6: Create)
CO4. **evaluate** the output of data analysis and machine learning models. (HOTS: Level 5: Evaluate)
CO5. **create** lab records of the solutions for the given assignments. (HOTS: Level 6: Create)
CO6. **demonstrate** use of ethical practices, self-learning and team spirit.. (LOTS: Level 3: Apply)

List of experiments/assignments:

1. Install Python and explore various popular IDE like IDLE, PyCharm, and Anaconda.
2. Assignments to perform various number operations like
    a. Find maximum from a list of numbers
    b. GCD of two number
    c. Square root of a number
    d. Check number is prime or not.
    e. Print first N prime numbers
    f. Remove duplicate numbers from list
    g. Print the Fibonacci series.
3. Assignments to perform various operations on Strings like creation, deletion, concatenation.
4. Create a List L = [10, 20, 30]. Write programs to perform following operations:
    a. Insert new numbers to list L.
    b. Delete numbers from list L. 99

c. Sum all numbers in list L.
d. Sum all prime numbers in list L.
e. Delete the list L.

5. Create a Dictionary D= {'Name': 'Allen', 'Age': 27, 5:123456}. Write programs to perform following operations:
   a. Insert new entry in D.
   b. Delete an entry from D.
   c. Check whether a key present in D.
   d. Update the value of a key.
   e. Clear dictionary D.
6. Two assignments on Sets to perform various operation like union, intersection, difference etc.
7. Two assignments related to searching operation like linear search, binary search.
8. Three assignments related to sorting like selection sort, bubble sort, insertion sort.
9. Demonstrate the use of dictionary for measuring student marks in five subjects and you have to find the student having maximum and minimum average marks.
10. Two assignment on usage of different available packages like random package to perform
    a. Print N random numbers ranging from 100 to 500.
    b. Print 10 random strings whose length between 3 and 5.
11. Two assignments on usage of package such as Numpy, Pandas.
12. Implement and demonstrate the functions of a simple calculator.
13. One assignment on implementing object oriented concept such as classes, inheritance, and polymorphism.
14. One assignment on file handling that how data is read and written to a file.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

MCS-38 Industrial/ Summer Training

General Course Information

Course Code: MCS-38 Course Credits: 2 Mode: Industrial Training	**Course Assessment Methods (100 Marks)** An internal evaluation is done by a faculty member appointed by the Head of the Department. Significance and originality of the problem addressed, and the solution provided: 20 Knowledge of the problem domain and tool used (VIVA-VOCE):25 Report Writing: 20 Judgement of the skill learnt, and system developed: 20 Level of ethics followed: 15

About the Industrial training:

Students do an Industrial Training of 4 to 6 weeks after second semester. They are expected to learn novel skills and develop some software application during the training period.

After doing training students will be able to:

CO1. **review** the existing systems for their strengths and weaknesses. (HOTS: Level 4: Analyse)
CO2. **address** novel problems in an original and innovative manner (HOTS: Level 6: Create)
CO3. **select and apply** modern software tools. (LOTS: Level 3: Apply)
CO4. **evaluate** the system developed critically with respect to the requirement analysis and other similar systems. (HOTS: Level 5: Evaluate)
CO5. **prepare** training report by organising ideas in an effective manner.
CO6. **follow** ethical practices while doing the training and writing report. (LOTS: Level 3: Apply)

MCS-41 IoT and Cloud Computing

General Course Information

Course Code: MCS-41 Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Three minor examinations (20 marks each) including third minor in open book mode will be conducted. The average of the highest marks obtained by a student in the any of the two minor examinations will be considered. Class Performance measurement through percentage of lectures attended (4 marks), assignments (6 marks), and the end-semester examination (70 marks) will be considered. For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basics of Computer Network, Distributed System.

About the Course: The objective of the course is to give students a comprehensive view understanding of the vision and impact of IoT and Cloud, cloud and IoT Market perspective and IoT and Cloud architecture and IoT.

Course Outcomes: By the end of the course students will be able to:
CO1. **define** concepts related to internet of things and cloud computing.
CO2. **analyse** cloud computing services used at various levels.
CO3. **apply** cloud computing techniques for various applications.
CO4. **express** the current status and expected future directions of the internet of things and cloud.
CO5. **identify** and propose applications which advance the cloud and IoT.
CO6. **develop** cloud based applications which advance the IoT.

Course Content

Unit I

Overview of Cloud Computing: Brief history and evolution - history of cloud computing, evolution of cloud computing, traditional vs. cloud computing, cloud service models (IaaS, PaaS & SaaS), cloud deployment models (public, private, hybrid and community cloud), benefits and challenges of cloud computing, introduction to AWS public cloud vendor, cost optimization in AWS, basics of virtualization, virtualization technologies, server virtualization, VM migration techniques, role of virtualization in cloud computing, introduction to EC2 service of AWS.

Unit II

Working with Private Cloud: Private cloud definition, characteristics of private cloud, private cloud deployment models, private cloud vendors - CloudStack, OpenStack, Eucalyptus Microsoft, private cloud ± benefits and challenges, private cloud implementation in Amazon EC2 service.

Unit III

Working with Public Clouds: What is public cloud, why public cloud, when to opt for public cloud, public cloud service models, public cloud players, infrastructure as a service offering, IaaS vendors, PaaS offerings, PaaS vendors, software as a service, demonstrating public cloud with AWS ± storage and database services, private vs. public cloud ± when to choose.

Unit IV

IoT Architecture- Introduction, State of the art: Architecture reference model introduction, reference model and architecture, IoT reference model. IoT Reference Architecture: IoT reference architecture-introduction, functional view, information view, deployment and operational view, other relevant architectural views, realworld design constraints- introduction, technical design constraints-hardware is popular again, data representation and visualization, interaction and remote control.

Text and Reference Books:

1. Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, "From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence", 1st Edition, Academic Press, 2014.

2. Hwang Kai, Fox Geoffrey C, Dongarra Jack G, "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Morgan Kaufmann Publishers, 2011.

3. Rittinghouse John W. and Ransome James F., "Cloud Computing: Implementation, Management, and Security", CRC Press,2009.

4. Velte Toby, Velte Anthony, Elsenpeter Robert, "Cloud Computing, A Practical Approach", TMH, 2013.

5. Vijay Madisetti and Arshdeep Bahga, "Internet of Things (A Hands-on- approach)", 1st Edition, VPT, 2014.

MCS-42 Data Warehouse and Data Mining

General Course Information

Course Code: MCS-42 Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Three minor examinations (20 marks each) including third minor in open book mode will be conducted. The average of the highest marks obtained by a student in the any of the two minor examinations will be considered. Class Performance measurement through percentage of lectures attended (4 marks), assignments (6 marks), and the end-semester examination (70 marks) will be considered. For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Knowledge of database systems, elementary knowledge of statistics and probability.

About the Course: Today's era is the era of information. Data is growing exponentially day by day. There is a need to process and analyse the data to extract knowledge from it, so that one can use that knowledge for decision making. This course provides introductory concepts of data mining and data warehousing. The course will be taught with a database as well as machine learning perspectives. The objective of the course is to provide a comprehensive understanding of data prep-processing, data mining tasks and evaluation of results obtained out of data mining processes.

Course Outcomes: By the end of the course students will be able to:

CO1. **understand** the various types of data mining and data warehouse concepts and techniques.
CO2. **explain** characteristics, architecture of a data warehouse, OLAP operations and data mining tasks.
CO3. **apply** various pre-processing and data mining techniques for extracting valuable information from data.
CO4. **evaluate** the descriptive and predictive data mining models.
CO5. **plan** a data mining process for discovering knowledge from real-world databases.

Course Content

Unit I

Introduction to Data Mining: Kind of data to be mined, Data Mining Functionalities, Technologies used in Data Mining, Applications of data Mining, Major Issues in Data Mining. Data Pre-Processing: Need for preprocessing, Data Objects and Attribute types, Statistical description of data, Data Visualization, Measuring similarity and dissimilarity of data, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization.

Unit II
Data Warehouse: Introduction, Data Warehouse and Database Systems, Data Warehouse Architecture, Data Warehouse Models, Data Cube and OLAP, Multidimensional data Model, Concept Hierarchies, OLAP operations, Data Warehouse Implementation

Unit III
Mining Associations and Correlations: Mining Frequent Patterns, Associations and Correlations, Frequent Itemset Mining using Apriori Algorithm, Generating Association Rules from Frequent Itemsets. Improving efficiency of Apriori, Pattern Growth Approach for Mining Frequent Itemsets, Pattern evaluation Methods. Advanced Pattern Mining: Pattern Mining in Multilevel and Multidimensional Space, ConstraintBased Frequent Pattern Mining.

Unit IV Classification: Introduction, Classification using Decision Tree Induction, Bayesian Classification Methods, Rule Based Classification, Model Evaluation and Selection, Techniques to Improve Classification Accuracy. Classification by Backpropagation, Support Vector Machines and Lazy Learners. Cluster Analysis: Introduction, Basic Clustering Methods, Partitioning Methods, Hierarchical Methods, Evaluation of Clustering.

Text and Reference Books:
1. Jiawei Han, Micheline Kamber and Jian Pei, Data Mining Concepts and Techniques, Morgan Kaufmann Publishers, Third Edition, July 2011.
2. Alex Berson, Stephen J. Smith, Data Warehousing, Data Mining & OLAP, Tata McGraw Hill, 2004.
3. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, Introduction to Data Mining, Pearson Education, 2014.
4. K. P. Soman, Shyam Diwakar and V. Ajay, Insight into Data Mining Theory and Practice, Easter Economy Edition, Prentice Hall of India,2009.
5. G. K. Gupta, Introduction to Data Mining with Case Studies, Prentice Hall of India, 2006.
6. Daniel T. Larose, Data Mining Methods and Models, Wiley, 2006.
7. W. H. Inman, Building the Data Warehouse, Wiley India, 2005.

MCS-43 Web Development

General Course Information

Course Code: MCS-43 Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Three minor examinations (20 marks each) including third minor in open book mode will be conducted. The average of the highest marks obtained by a student in the any of the two minor examinations will be considered. Class Performance measurement through percentage of lectures attended (4 marks), assignments (6 marks), and the end-semester examination (70 marks) will be considered. For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: knowledge of Computer Basics

About the Course: Web development is a management of information. Web Development is a core and an essential course for every graduate in Computer Science and Engineering. This course introduces web designing tools like HTML, XML, Java Script and ASP/JSP etc. and various web site will be designed with the help of these tools for solving real world problems. It includes various types of website. Further, It is more useful for dynamic programming as well.

Course Outcomes: By the end of the course students will be able to:

CO 1. **enlist** principles of Information Architecture for Web design. (LOTS: Level 1: Remember)

CO 2. **explain** navigational systems, labeling systems, and taxonomies for websites. (LOTS: Level 2: Understand)

CO 3. **apply** basic web designing tools (HTML, XML, ASP/JSP, JQuery, Java Script). (LOTS: Level 3: Apply)

CO 4. **evaluate** critically design of webpages based on various technologies. (HOTS: Level 5: Evaluate)

CO 5. **create** a report describing or making recommendations for a website design. (HOTS: Level 6: Create)

Course Content

Unit I

Information Architecture, Role of Information Architect, Collaboration and Communication, Organizing Information, Organizational Challenges, Organizing Web Sites and Intranets, Creating Cohesive Organization Systems Designing, Navigation Systems, Types of Navigation Systems, Integrated Navigation Elements, Remote Navigation Elements, Designing Elegant Navigation Systems, Searching Systems, Designing the Search Interface, Indexing the Right Stuff, What to Search or not to Search, Grouping Content, Conceptual Design, Architecture Blueprints, Architectural Page Mockups, Design Sketches.

Unit II

Structured Information, Design and Documentation, XML Web 6.0, JDBC, Metadata, Unstructured Information, Techniques for Unstructured Information, HTML Basic Concepts, Good Web Design, Process of Web Publishing, Phases of Web Site Development, Structure of Html Documents, Html Elements for Designing Pages. Text Level Events, Linking Basics, Linking In Html, Images and Anchors Attributes, Image Maps, Semantic Linking Meta Information, Image Preliminaries, Images, Layout Design, Advanced Layout. Audio Support in Browsers, Video Support, Other Binary Formats. Style Sheets, Positioning with Style Sheets. Basic Interactivity and Html: Forms, Forms Control, Advance HTML and Web Designing.

Unit III

Alternative Technologies for Designing, The Hypertext Transport Protocol, URLs, HTTP, Browser Requests, Server Responses, Proxies, Content Negotiation, The Common Gateway Interface, The CGI Environment Variables. CGI Output, Forms and CGI, Sending Data to the Server, Form Tags, Decoding Form Input, Architectural Guidelines, Coding Guidelines, Efficiency and Optimization. JSP Basics, Integrating Scripts in JSPs, ASP Objects and Components, JSP: Request and Response Objects, Retrieving the Contents of a HTML form, retrieving a Query String, Cookies, Creating and Reading Cookies.

Unit IV

XML basics, Relationship between HTML, SGML, and XML, Valid Documents. Ways to use XML, XML for Data Files, Embedding XML into HTML documents, Converting XML to HTML for DISPLAY, Displaying XML using CSS and XSL, Rewriting HTML as XML, Basics of Advance Web Development Tools.

Text and Reference Books:

1. Thomas A Powell, HTML-The Complete Reference, Tata McGraw Hill, 2003.
2. Scott Guelich, Shishir Gundavaram, Gunther Birzniek, CGI Programming with Perl 2 nd edition O'Reilly, 2000.
3. Doug Tidwell, James Snell, Pavel Kulchenko, Programming Web Services with SOAP, O'Reilly 2009.
4. Young, XML Step by Step, 2nd edition, PHI.
5. Paul J. Deitel, Harvey M. Deitel, Abbey Deitel, Internet & World Wide Web How to Program, 5t edition, 2008.

MCS-44 (i) Software Project Management
(Elective Course -III)

General Course Information

Course Code: MCS-44 (i) Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Three minor examinations (20 marks each) including third minor in open book mode will be conducted. The average of the highest marks obtained by a student in the any of the two minor examinations will be considered. Class Performance measurement through percentage of lectures attended (4 marks), assignments (6 marks), and the end-semester examination (70 marks) will be considered. For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Preliminary knowledge of Software Engineering.

About the Course: The course involves training students in software project management and project planning. It focuses on the need for careful planning, monitoring and control for delivering quality projects in time. Besides this student learn to measure the success of a project in meeting its objectives.

Course Outcomes: By the end of the course students will be able to:

CO1. **study** the basic concepts related to stepwise project planning. (LOTS: Level 1: Remember)
CO2. **demonstrate** the knowledge about Quality Control, Standard and Risk Management. (LOTS: Level 2: Understand)
CO3. **illustrate** the Activity Planning, and Resource Allocation Process. (LOTS: Level 2: Understand)
CO4. apply the concept of team structure and organization structure. (LOTS: Level 3: Apply)
CO5. **compare** various Project Evaluation and Estimation Techniques. (HOTS: Level 4: Analyse)
CO6. **plan** activities necessary for completing the software projects successfully. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction to Software Project Management (SPM): Definition of Software Project, Software Project Vs Other types of projects, activities covered by SPM, categorizing software projects, project as system, management control, Requirement specification, Information and control in organization, project management lifecycle. Stepwise Project Planning: Introduction, selecting a project, identifying project scope and objectives, identifying project infrastructure, analysing project characteristics, identifying the project products and activities, estimate efforts for each activity, identifying activity risk, allocate resources, review/publicize plan.

Unit II

Project Evaluation and Estimation: Cost-Benefit analysis, cash flow forecasting, cost benefit evaluation techniques, Selection of an appropriate project, choosing technologies, choice of process models, rapid application development, waterfall model, V process model and spiral model, Albrecht function point analysis. Activity Planning: Objectives of activity planning, project schedule, projects and activities, sequencing and scheduling activities, network planning model.

Unit III

Risk Management: Introduction, the nature of risk, managing risk, risk identification, risk analysis, reducing the risks, evaluating risks to schedule, calculating z-values. Resource Allocation: Introduction, the nature of resources, identifying resource requirements, scheduling resources, creating critical paths.

Unit IV

Managing Contracts and People: Introduction, types of contract, stages in contract placement, terms of contract, contract management, acceptance, managing people and organizing teams: Introduction, understanding organization behaviour: a back ground, selecting the right person for job, instruction in best methods, motivation, working in groups, becoming a team, decision making, leadership, organization structures. Software Quality: Introduction, the place of software quality in project planning, the importance of software quality, defining software quality, McCall's software quality factors, product versus process quality management, external standards, techniques to enhance software quality.

Text and Reference Books:

1. Bob Hughes and Mike Cotterell , Software Project Management, Sixth Edition, TMH, 2018.
2. Walker Royce , Software Project Management, Addison Wesley, 1998.
3. Pankaj Jalote , Software Project Management in Practice, Pearson, 2002.
4. Ramesh, Managing Global Software Projects, TMH, 2005.

MCS-44 (ii) Big Data Analytics
(Elective Course -III)

General Course Information

Course Code: MCS-44 (ii) Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Three minor examinations (20 marks each) including third minor in open book mode will be conducted. The average of the highest marks obtained by a student in the any of the two minor examinations will be considered. Class Performance measurement through percentage of lectures attended (4 marks), assignments (6 marks), and the end-semester examination (70 marks) will be considered. For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basics of statistics and data mining.

About the Course: This course aims to provide students with the knowledge of current challenges, methodologies and technologies in processing big data. Emphasis will be placed on the students' understanding of the rationales behind the technologies and the students' ability to analyse big data using professional packages and tools.

Course Outcomes: By the end of the course students will be able to:

CO1. **recall** the concepts of big data analysis. (LOTS: Level 1: Remember)

CO2. **interpret** the outcomes of big data analysis. (LOTS: Level 2: Understand)

CO3. **apply** technical skills and modern tools for descriptive and predicative modelling. (LOTS: Level 3: Apply)

CO4. **analyse** a framework for visualization of big data analytics for business user. (HOTS: Level 4: Analyse)

CO5. **examine** critically the results of mining to support business decision-making. (HOTS: Level 5: Evaluate)

CO6. **design** schemes for big data analytics for solving big data problems in efficient manner. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction: Overviews of Big Data, State of the Practice in Analytics, The Data Scientist, Big Data Analytics in Industry Verticals, Data Analytics Lifecycle Challenges of Conventional Systems, Statistical Concepts: Sampling Distributions, Re-Sampling, Statistical Inference, Prediction Error, Regression Modelling, Multivariate Analysis, Bayesian Modelling.

Unit II
Mining Data Streams: Stream Data Model and Architecture, Stream Computing, Sampling Data in a Stream, Filtering Streams, Counting Distinct Elements in a Stream, Estimating Moments, Counting Oneness in a Window, Decaying Window, Real time Analytics, Platform (RTAP) Applications, Case Studies, Real Time Sentiment Analysis, Stock Market Prediction

Unit III
Frequent Itemset and Clustering: Mining Frequent Itemsets, Market Based Model: Apriori Algorithm, Handling Large Data Sets in Main Memory, Limited Pass Algorithm, Counting Frequent Itemsets in a Stream, Clustering based Techniques: Hierarchical, K-Means etc., Clustering High Dimensional Data, CLIQUE And PROCLUS, Frequent Pattern based Clustering Methods, Clustering in Non-Euclidean Space, Clustering for Streams and Parallelism.

Unit IV
Frameworks and Visualization: Overview of MapReduce, Hadoop, Hive, MapR, Sharding, NoSQL Databases, S3, HADOOP, Distributed File System (HDFS), Visualizations: Visual Data Analysis Techniques, Interaction Technique and Applications.

Text and Reference Books:
1. Michael Berthold, David J. Hand, Intelligent Data Analysis, Springer, 2007.
2. A. Rajaraman, J.D. Ullman, Mining of Massive Datasets, Cambridge University Press, 2012.
3. Bill Franks, Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics, John Wiley & sons, 2012.
4. Glenn J. Myatt, Making Sense of Data, John Wiley & Sons, 2007
5. Pete Warden, Big Data Glossary, O'Reilly, 2011.

MCS-44 (iii) Cyber Security
(Elective Course -III)

General Course Information

Course Code: MCS-44 (iii) Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Three minor examinations (20 marks each) including third minor in open book mode will be conducted. The average of the highest marks obtained by a student in the any of the two minor examinations will be considered. Class Performance measurement through percentage of lectures attended (4 marks), assignments (6 marks), and the end-semester examination (70 marks) will be considered. For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Prerequisite: Computer networks, essentials of computer security, programming languages like python, java etc.

About the Course: The increase in techniques for unauthorized access into systems has led to variety of cyber- attacks. To mitigate the exploitation of the vulnerabilities leading to these attacks, we need to adopt robust security architecture into our premises. We have to choose between various cyber security technologies. In the current scenario, we require to secure end-to-end devices, networks, networking devices. The objective of this course is to enable students to get acquainted to cyber security principles to be followed while working online and offline.

Course Outcomes: By the end of the course students will be able to:

CO1. **recognize** the terminology associated with cyber security.
CO2. **represent** cyber security in terms of ethics, principles, Intellectual property and Trademarks.
CO3. **analyze** cyber activities on the internet to follow IT Act.
CO4. **evaluate** cybercrime situations and recommend appropriate cyber security laws
CO5. **integrate** frameworks to sustain critical infrastructures.

Course Contents

Unit I

Cyber Security Fundamentals: Network and Security Concepts: Firewalls, Virtualization, DNS, Radio-Frequency Identification, Attacker Techniques and Motivations: Tunneling Techniques, Fraud Techniques, Threat Infrastructure, Exploitation, Malicious code. Defense and Analysis Techniques.

Unit II

Ethics in Cyber Security: Privacy, Intellectual property in the cyberspace, Professional ethics, Freedom of speech, Fair user and ethical hacking, Trademarks, Internet fraud, Electronic evidence, Forensic

technologies, Digital evidence collections. Tools and methods used in cybercrime: Introduction, Password cracking, Keyloggers and spywares, Virus and worms, Phishing and identity theft, Trojan horses and backdoors, Steganography.

Unit III

Cyber crimes and Cyber security: Cyber crime and legal landscape around the world, Cyber laws, The Indian IT Act, Challenges, Digital signatures and Indian IT Act, Amendments to the Indian IT Act, Cyber crime and punishment, Cost of Cyber crimes and IPR Issues, Web threats for organizations, Social computing and associated challenges for organizations.

Unit IV

Protecting Critical Infrastructures: Critical Infrastructures: Key Assets, Critical Infrastructure Interdependencies, Internet, Social Media and Cyber Attacks on Critical Infrastructures, Cyber Threat Spectrum- Cyberspace Attacks and Weapons, Framework for improving Critical Infrastructure Cyber security.

Text and Reference Books:

1. James Graham, Richard Howard, "Cyber Security Essentials", CRC Press, Taylor & Francis Group, ISBN: 978-1-4398-5126-5, 2011.
2. Thomas A. Johnson, "Cyber-Security Protecting Critical Infrastructures from Cyber Attack and Cyber Warfare", CRC Press, ISBN:978-1-4822-3923-2, 2015.
3. Nina Godhole and Sunit Belapure, Cyber Security, Wiley India, 2011.

MCS-45 (i) Soft Computing
(Elective Course -IV)

General Course Information

Course Code: MCS-45 (i) Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Three minor examinations (20 marks each) including third minor in open book mode will be conducted. The average of the highest marks obtained by a student in the any of the two minor examinations will be considered. Class Performance measurement through percentage of lectures attended (4 marks), assignments (6 marks), and the end-semester examination (70 marks) will be considered. For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Prerequisites: Basic knowledge of Probability Theory, Set Theory and, Data Structure and Computer Algorithms.

About the Course: We need to learn soft computing techniques to make intelligent machines that possess human like abilities to reason, learn and handle the uncertainty and vagueness often inherent in real world problems. Unlike conventional computing, soft computing techniques are tolerant of imprecision, uncertainty and approximations, and provide low cost, robust and tractable solutions to the complex real-world problems where conventional methods fail to do so. This introductory course on soft computing is going to cover Genetic Algorithms, Artificial Neural Networks and Fuzzy Logic.

Course Outcomes: By the end of the course students will be able to:

CO1. **define** the terminology and concepts related to soft computing techniques.
CO2. **discuss** soft computing techniques including genetic algorithms, fuzzy systems and neural networks.
CO3. **solve** problems related to Genetic algorithms, Fuzzy logic and Neural Networks.
CO4. **analyse** the design of Genetic Algorithms, Neural Networks and Fuzzy Systems.
CO5. **justify** the design of a soft computing algorithm for a given problem.
CO6. **design** Genetic Algorithms and Neural Networks to solve optimization and pattern recognition problems.

Course Content

Unit I

Introduction to Soft Computing and related definitions: Defining soft computing, Differentiating the situations for application of hard and soft computing; Working of a simple Genetic Algorithm: Representation/Encoding 149 Schemes, initializing a GA population, evaluation function, genetic

operators, Function optimization using GA. Study of parameters of genetic algorithms and its performance, sampling and selection mechanisms. Scaling of GA population.

Unit II

Designing Genetic Algorithms for different applications: Different types encoding schemes, role of fitness function, different types of genetic operators, Designing GAs for numerical optimization, knapsack problem and travelling salesperson and other similar problems.

Unit III

Fuzzy sets: Basic terminology and definitions, Operations on Fuzzy sets, MF formulations and parameterisation, MFs of one and two dimensions, Derivatives of parameterised MFs, Fuzzy numbers, Extension principle and fuzzy relations, Operations on Fuzzy relations, Linguistic variables, Fuzzy If-Then Rules, Compositional rule of inference.

Unit IV

Neural networks: Basic terminology and definitions, Model of an artificial neuron, Sigmoid function, Neural Network Architectures, Rosenblatt's Perceptron, Fixed increment perceptron learning algorithm for a classification problem, Examples of learning of AND/OR gate by perceptron, XOR problem. Back Propagation Neural Networks: Architecture of a backpropagation network, Model for multi-layer perceptron, Back propagation learning, Delta or gradient descent learning rule and effect of learning rate, Back propagation learning algorithm.

Text and Reference Books:

1. David. E. Goldberg, Genetic Algorithms in Search, Optimization and machine learning, Addison Wesley, 1999.
2. Zbigniew Michalewicz, Genetic algorithms + Data Structures = Evolution Programs, Springers-Verlag, 1999.
3. M. Mitchell, An Introduction to Genetic Algorithms, Prentice-Hall, 1998.
4. S. Rajasekaran & G. A. Vijayalakshmi Pai, Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis & Applications, PHI, 2003.
5. S. N. Sivanandam & S. N. Deepa, Principles of Soft Computing, Wiley - India, 2007.
6. J-S. R. Jang, C.-T. Sun, E. Mizutani, Neuro-Fuzzy and Soft Computing, PHI, 1997.
7. Simon O. Haykin, Neural Networks, A Comprehensive Foundation, PHI, 1994.

MCS-45 (ii) Compiler Design
(Elective Course -IV)

General Course Information

Course Code: MCS-45 (ii) Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Three minor examinations (20 marks each) including third minor in open book mode will be conducted. The average of the highest marks obtained by a student in the any of the two minor examinations will be considered. Class Performance measurement through percentage of lectures attended (4 marks), assignments (6 marks), and the end-semester examination (70 marks) will be considered. For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Brief knowledge of programming languages, Data Structure, and Algorithm Design.
About the Course: Compilers have become part and parcel of today's computer systems. These are responsible for making the user's computing requirements, specified as a piece of program, understandable to the underlying machine. There tools work as interface between the entities of two different domains – the human being and the machine. The actual process involved in this transformation is quite complex. Compiler design covers basic translation mechanism and, error detection and recovery. It includes lexical, syntax, and semantic analysis as front end, and code generation and optimization as back-end.

Course Outcomes: By the end of the course students will be able to:
CO1. **state** principles of compiler design. (LOTS: Level 1: Remember)
CO2. **illustrate** the essential phases for automatically converting source code into object code. (LOTS: Level 2: Understand)
CO3. **apply** lexical analysis, syntax analysis and code optimization techniques for solving problems. (LOTS: Level 3: Apply)
CO4. **analyse** a parse tree and a given BNF grammar. (LOTS: Level 4: Analyse)
CO5. **compare** and contrast syntax-oriented translation schemes (HOTS: Level 5: Evaluate)
CO6. **design** a lexical analyser from the specification of a language's lexical rules. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction To Compilers: Compilers and translators, need of translators, structure of compiler its different phases, Compiler construction tools. Lexical Analysis: Role of lexical analyzer, design of lexical analyzer, regular expressions, Specification and recognition of tokens, input buffering, A

language specifying lexical analyzer. Finite automata, conversion from regular expression to finite automata, and vice versa, minimizing number of states of DFA, Implementation of lexical analyzer.

Unit II

Syntax Analysis: Role of parsers, context free grammars, definition of parsing. Parsing Technique: Shiftreduce parsing, operator precedence parsing, top down parsing, predictive parsing.

Unit III

LR parsers, SLR, LALR and Canonical LR parser. Syntax Directed Translations: Syntax directed definition, construction of syntax trees, syntax directed translation scheme, implementation of syntax directed translation, three address code, quadruples and triples.

Unit IV

Symbol Table & Error Detection and Recovery: Symbol tables, its contents and data structure for symbol tables; trees, arrays, linked lists, hash tables. Errors, lexical phase error, syntactic phase error, semantic error. Code Optimization & Code Generation: Code generation, forms of objects code, machine dependent code, optimization, register allocation for temporary and user defined variables.

Text and Reference Books:

1. Alfread V. AHO, Ravi Sethi and J.D. Ullman, Compilers Principle, Techniques and Tools, Addison Wesley, 2007.
2. Tremblay and Sorenson, Theory and practice of compiler writing, Mc. Graw Hill, 1985.
3. Dhamdare, System software, MGH, 1986.
4. Alfred V. Aho, Jeffrey D. Ullman, Principles of Compiler Design, Narosa Publication, 2002.

MCS-45 (iii) Mobile Application Development
(Elective Course -IV)

General Course Information

Course Code: MCS-45 (iii) Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Three minor examinations (20 marks each) including third minor in open book mode will be conducted. The average of the highest marks obtained by a student in the any of the two minor examinations will be considered. Class Performance measurement through percentage of lectures attended (4 marks), assignments (6 marks), and the end-semester examination (70 marks) will be considered. For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Java Programming and Object-Oriented programming, Knowledge of RDBMS and OLTP.

About the Course: Mobile Application Development has been introduced as a Professional Elective course for Students keeping in view the Employers' requirements. Android Platform forms the basis for developing Mobile Applications since the last decade as compared to IOS Platform for Apple Products. The Environment requires User Interface to be developed using Buttons, Check-Boxes, Alert Dialog and its kind.

Course Outcomes: By the end of the course students will be able to:

CO1. **state** basic of Android , its Evolution and its Architecture. (LOTS: Level 1: Remember)

CO2. **demonstrate** the Lifecycle of Software for Android Mobile Applications. (LOTS: Level 2: Understand)

CO3. **prepare** Mobile Applications on the Android Platform. (LOTS: Level 3: Apply)

CO4. **compare** working with Buttons and other Widgets for Visual Environment. (HOTS: Level 4: Analyse)

CO5. **develop** Mobile Applications using data storage in SQLite Database and evaluate its Performance. (HOTS: Level 6: Create)

Course content

Unit I

Mobile OS Architecture: Android, Blackberry OS, Firefox OS, IOS, Window OS, ARM and MIPS processor, Challenges of the mobile platform, Hello Android example, Internal Details, Dalvik VM, Software Stack, Android Core Building Blocks, Android Emulator, AndroidManifest.xml, R.java file, Hide Title Bar, Screen Orientation.

Unit II

UI Widgets: Working with Button, Toast, Custom Toast, Button, Toggle Button, Switch Button, Image Button, CheckBox, Alert Dialog, Spinner, AutoCompleteTextView, RatingBar, DatePicker, TimePicker, ProgressBar, Quick Contact Budge, Analog Clock and Digital Clock, Working with hardware Button, File Download.

Unit III

Activity, Intent & Fragment: Activity Lifecycle, Activity Example, Implicit Intent, Explicit Intent, Fragment Lifecycle, Fragment Example, Dynamic Fragment. Android Menu: Option Menu, Context Menu, Popup Menu Layout Manager: Relative Layout, Linear Layout, Table Layout, Grid Layout.

Unit IV

Adaptor: Array Adaptor, ArrayList Adaptor, Base Adaptor. View: GridView, WebView, ScrollView, SearchView, TabHost, DynamicListView, Expanded ListView. SQLite: SQLite API, SQLite Spinner, SQLite ListView XML & JSON: XML Parsing SAX, XML Parsing DOM, XML Pull Parser, JSON basics, JSON Parsing.

Text and Reference Books:

1. Redazione Io Programmo, Android Programming, 2011
2. John Horton, Android Programming for Beginners, packt publishing, 2015
3. Jason Wei, Android Database Programming, packt publishing, 2012
4. Mark L Murphy, Android Programming Tutorials, 3rd Edition, 2010
5. Bill Phillips et al., Android Programming - The "Big Nerd Ranch" Guide 2017
6. Rick Rogers et al., Android Application Development: Programming with the Google SDK, 2009.

D.

MCS-46 IoT and Cloud Computing Lab.

General Course Information

Course Code: MCS-46 Course Credits: 2 Type: Professional Core Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external:70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner (appointed by the Controller of Examination) in association with the internal examiner appointed by the Head of the Department.

Pre-requisites: Basic programming skills.

About the Course: This lab. course on IoT and Cloud Computing helps students to learn how to use cloud services, implement virtualization and task scheduling, apply the vision of IoT and understand IoT in applied form.

Course Outcomes: By the end of the course students will be able to:

CO1. **analyse** the cloud computing setup with its vulnerabilities and applications using different architectures.

CO2. **design** different workflows according to requirements and apply map reduce programming model.

CO3. **identify** and propose applications which advance the IoT.

CO4. **develop** applications which advance the IoT.

CO5. **create** lab record for assignments that includes problem definitions, design of solutions and conclusions.

CO6. **demonstrate** use of ethical practices, self-learning and team spirit.

List of experiments/assignments:

1. Amazon Simple Storage Service (Amazon S3) and Amazon Glacier Storage
2. Amazon Elastic Compute Cloud (Amazon EC2) and Amazon Elastic Block Store
3. Amazon Virtual Private Cloud (Amazon VPC)
4. Elastic Load Balancing, Amazon CloudWatch, and Auto Scaling
5. AWS Identity and Access Management (IAM)
6. Databases and AWS
7. SQS, SWF, and SNS
8. Domain Name System (DNS) and Amazon Route
9. Amazon ElastiCache
10. Additional Key Services
11. Security on AWS
12. MQTT, REST/HTTP, CoAP, MySQL, apache for handling HTTP Requests, PHP & MySQL for data processing, MongoDB object type database
13. HTML, CSS & jQuery for UI designing, JSON lib for data processing, security & privacy during development, Working with arduino and intel galileo boards/Rsapberry Pi

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

MCS-47 Project Work

General Course Information

Course Code: MCS-47 Course Credits: 4 Mode: Design and development of project in lab. No. of hours per week: 4	**Course Assessment Methods (Internal evaluation: 30 marks; External Evaluation marks: 70)** Evaluation is done by the internal examiner (project guide) and external examiner appointed by Controller of Examination. The criteria for evaluation are given below. 1.Review of literature related to problem domain: 4(In.) +10(Ex.) 2.Significance and originality of the solution presented: 9(In.) +20(Ex.) 3.Significance and Scope of results: 9(In.) +20(Ex.) 4.Organisation and presentation of minor project report: 4(In.) +15(Ex.) 5. Level of Ethics followed: 4(In.) +5(Ex.)

About the Project Work:

Students start working on their project work in the beginning of fourth semester. Students do the background research for identifying appropriate problems, methodology and tools for their respective project works. They prepare a synopsis of the project work to be carried out. Each student is required to prepare a synopsis in the format provided and present it in front of a committee constituted by the Head of the Department. Students can carry out projects in groups of two. In case of group project, the size of the problem should be significant, and members of the group must specify their individual contribution.

After approval by the internal committee, they continue working on their project work throughout 4th semester. They carry out implementation of their respective projects based on the problem identified, methodology and tools suggested in the approved synopsis. They are required to complete their project work by the end of 4th semester. They prepare the final project reports according to the format provided. At the end of fourth semester, each student is required to present his/her project work in front of internal project guide and external examiner appointed by Controller of Examination.

Course Outcomes: After doing Project students will be able to:

CO1. **evaluate** critically the existing solutions and methodologies through reviewing literature.
CO2. **organize** and communicate (written and oral) ideas effectively.
CO3: **plan** the project according to principles of project management.
CO4. **devise** original solutions to complex problems using modern tools.
CO5. **justify** the outcomes of the project work.
CO6. **develop** solutions that meet ethical, societal and legal considerations.