

Masters in Computer Science (Semester System)

M.Sc.(Computer Science)

**Session 2022-23
Exam 2022 and 2023**

**Department of Computer Science
Maharaja Ganga Singh University, Bikaner**

Masters in Computer Science (Semester System)

Table of Contents

S.No.	Item	Page No
1	Background	3
2	Programme Outcomes (POs)	5
3	Programme Specific Outcomes (PSOs)	7
4	Post Graduate Attributes	8
5	Structure of Masters' Courses	9
6	Learning Outcome Index	13
7	Semester-wise Courses & Credit Distribution	14
8	Course Level Learning Outcomes	14
9	Teaching-Learning Process	72
10	Assessment & Evaluation	72

Masters in Computer Science (Semester System)

Background

Considering the curricular reforms as instrumental for desired learning outcomes, all the academic departments of Maharaja Ganga Singh University Bikaner, made a rigorous attempt to revise the curriculum of postgraduate programs in alignment with National Education Policy-2020 and UGC Quality Mandate for Higher Education Institutions-2021. The process of revising the curriculum could be prompted with the adoption of the “Comprehensive Roadmap for Implementation of NEP-2020”. The Roadmap identified the key features of the Policy and elucidated the Action Plan with well-defined responsibilities and an indicative timeline for major academic reforms.

The process of revamping the curriculum started with a series of webinars and discussions conducted by the University to orient the teachers about the key features of the Policy, enabling them to revise the curriculum in sync with the Policy. Proper orientation of the faculty about the vision and provisions of NEP-2020 made it easier for them to appreciate and incorporate the vital aspects of the Policy in the revised curriculum focused on creating holistic thoughtful, creative, and well-rounded individuals equipped with the key 21st-century skills ‘for the development of an enlightened, socially conscious, knowledgeable, and skilled nation’.

With NEP-2020 in the background, the revised curricula articulate the spirit of the Policy by emphasising upon - an integrated approach to learning; innovative pedagogies and assessment strategies; multidisciplinary and cross-disciplinary education; creative and critical thinking; ethical and Constitutional values through value-based courses; 21st century capabilities across the range of disciplines through life skills, entrepreneurial and professional skills; community and constructive public engagement; social, moral, and environmental awareness; Organic Living and Global Citizenship Education (GCED); holistic, inquiry-based, discovery-based, discussion-based and analysis-based learning; exposure to Indian knowledge system, cultural traditions and literature through relevant courses offering “Knowledge of India, fine blend of modern pedagogies with indigenous and traditional ways of learning; flexibility in course choices, student-centric participatory learning; imaginative and flexible curricular structures to enable creative combinations of disciplines for study; offering multiple entry and exit points, alignment of Vocational courses with the International Standard Classification of Occupations maintained by the International Labor Organization; breaking the silos of disciplines; integration of extra-curricular and curricular aspects, exploring internships with local industry, businesses and artists and craft persons; closer collaboration between industry and higher education institutions for technical, vocational, and science programs, and formative assessment tools to be aligned with the learning outcomes, capabilities, and dispositions as specified for each course. The university has also developed a consensus on Blended Learning with 10% component of online teaching and 60% face-to-face classes for each program.

The revised curricula of various programs could be devised with concerted efforts of the faculty, Heads of the Departments, and the Deans of Schools of Study. The draft prepared by each department was discussed in a series of discussion sessions conducted at the Department, School, and University level. The leadership of the University has been a driving force behind the entire exercise of developing the uniform template and structure for the revised curriculum. The Vice-Chancellor of the University conducted series of meetings with Heads and Deans to deliberate upon the vital parameters of the revised curriculum to formulate a uniform template featuring Background, Programme Outcomes, Programme

Masters in Computer Science (Semester System)

Specific Outcomes, Postgraduate Attributes, Structure of Masters Course, Learning Outcome Index, Semester-wise Courses and Credit Distribution, Course-level Learning Outcomes, Teaching-Learning Process, Blended Learning, Assessment and Evaluation, Keywords, References, and Appendices. The experts of various Board of Studies and School Boards contributed to a large extent in giving the final shape to the revised curriculum of each program.

To ensure the implementation of curricular reforms envisioned in NEP-2020, the University has decided to implement various provisions in a phased manner. Therefore, the curriculum may be reviewed annually so as to gradually include all relevant provisions of NEP-2020.

Masters in Computer Science (Semester System)

Program Outcomes

On completing Masters in the Faculty of Science, the students shall be able to realize the following outcomes:

PO	Description
PO1	Acquired knowledge with facts and figures related to various subjects in pure sciences such as Physics, Chemistry, Botany, Zoology, Mathematics, etc.
PO2	Understood the basic concepts, fundamental principles, and scientific theories related to various scientific phenomena and their relevance in day-to-day life.
PO3	Acquired the skills in handling scientific instruments, planning, and performing laboratory experiments The skills of observations and drawing logical inferences from the scientific experiments.
PO4	Analyzed the given scientific data critically and systematically and the ability to draw objective conclusions.
PO5	Been able to think creatively (divergent and convergent) to propose novel ideas in explaining facts and figures or providing new solutions to problems.
PO6	Realized how developments in any science subject help develop other science subjects and vice-versa and how interdisciplinary approach helps provide better solutions and new ideas for sustainable outcomes.
PO7	Developed a scientific outlook concerning science subjects and all aspects related to life.
PO8	Realized that knowledge of subjects in other faculties such as humanities, performing arts, social sciences, etc., can have greatly and effectively influence, which inspires in evolving new scientific theories and inventions.
PO9	Imbided ethical, moral, and social values in personal and social life, leading to a highly cultured and civilized personality.
PO10	Developed various communication skills such as reading, listening, speaking, etc., which will help express ideas and views clearly and effectively.
PO11	Realized that pursuit of knowledge is a lifelong activity and in combination with untiring efforts and positive attitude and other necessary qualities leads towards a successful life.

Masters in Computer Science (Semester System)


Program Specific Outcomes (PSO)

On completing Masters in the M.Sc. in Computer Science, the students shall be able to realize the following outcomes:

PSO	Description
PSO1	Communicate computer science concepts, designs, and solutions effectively and professionally
PSO2	Apply knowledge of computing to produce effective designs and solutions for specific problems
PSO3	Use software development tools, software systems, and modern computing platforms
PSO4	To have the knowledge and the ability to develop creative solutions
PSO5	To develop skills to learn new technology
PSO6	To develop critical reasoning
PSO7	To apply computer science theory and software development concepts to construct computing-based solutions
PSO8	To design and develop computer programs/computer-based systems in the area related to algorithms, networking, web design, cloud computing, Artificial Intelligence, Mobile applications
PSO9	The ability to understand, analyse and develop computer programs in the areas related to algorithms, system software, multimedia, web design, big data analytics, and networking for efficient design of computer-based systems of varying complexity
PSO10	The ability to understand the evolutionary changes in computing, apply standard practices and strategies in software project development using open-ended programming environments to deliver a quality product for business success, real-world problems, and meet the challenges of the future
PSO11	The ability to employ modern computer languages, environments, and platforms in creating innovative career paths to be an entrepreneur, lifelong learning and a zest for higher studies and also to act as a good citizen by inculcating in them moral values & ethics

Postgraduate Attributes

Masters in Computer Science (Semester System)

- Disciplinary Knowledge
 - Creative & Critical Thinking
 - Reasoning and Analytical abilities
 - Logic/Discrete Mathematics knowledge
 - Logical Thinking
 - Problem analysis and solving abilities
 - Life Skills
 - Moral & Ethical Values
 - Research Skills
- 

Masters in Computer Science (Semester System)

Structure of Masters' Programme

Scheme for
M.Sc. Computer Science (Semester I)
Examination 2022
Session 2022-23

Semester I										
	Course Code	Course Title	Exam Hours	Max. Marks		Min. Marks	L	T	P*	Credits
				Int. Marks	Ext. Marks					
Core Courses										
1	FS-COMP- MSC-CS-CC- 101	Mathematics for Computer Science	3	10	40	13 (25%)	3	1	1	5
2	FS-COMP- MSC-CS-CC- 102	Internet Programming	3	10	40	13 (25%)	3	1	1	5
3	FS-COMP- MSC-CS-CC- 103	Computer Organization	3	10	40	13 (25%)	3	1	1	5
4	FS-COMP- MSC-CS-CC- 104	C++ Programming	3	10	40	13 (25%)	3	1	1	5
5	FS-COMP- MSC-CS-CP- 105	Combined Practical	3	25	75	26 (25%)	*combined practical of above subjects			

Masters in Computer Science (Semester System)

**Scheme for
M.Sc. Computer Science (Semester II)
Examination 2023
Session 2022-23**

Semester II										
	Course Code	Course Title	Exam Hours	Max. Marks		Min. Marks	L	T	P*	Credits
				Int. Marks	Ext. Marks					
Core Courses										
1	FS-COMP-MSC-CS-CC-201	Database Management System	3	10	40	13 (25%)	3	1	1	5
2	FS-COMP-MSC-CS-CC-202	Data Communication and Networking	3	10	40	13 (25%)	3	1	1	5
3	FS-COMP-MSC-CS-CC-203	Operating System	3	10	40	13 (25%)	3	1	1	5
4	FS-COMP-MSC-CS-CC-204	PHP	3	10	40	13 (25%)	3	1	1	5
5	FS-COMP-MSC-CS-CP-205	Combined Practical	3	25	75	26 (25%)	*combined practical of above subjects			

Masters in Computer Science (Semester System)

**Scheme for
M.Sc. Computer Science (Semester III)
Examination 2023
Session 2022-23**

Semester III										
	Course Code	Course Title	Exam Hours	Max. Marks		Min. Marks	L	T	P*	Credits
				Int. Marks	Ext. Marks					
Core Courses										
1	FS-COMP-MSC-CS-CC-301	Data Structures	3	10	40	13 (25%)	3	1	1	5
2	FS-COMP-MSC-CS-CC-302	Java	3	10	40	13 (25%)	3	1	1	5
Core Elective Courses										
3	FS-COMP-MSC-CS-CE-303	a) Software Engineering & Research Methodology b) Artificial Intelligence c) Python d) Theory of Computation	3	10	40	13 (25%)	3	1	1	5
4	FS-COMP-MSC-CS-CP-305	Combined Practical	6	25	75	26 (25%)	*combined practical of above subjects			
Open Elective Course										
1	FS-COMP-MSC-CS-EO-306	a) Data Analysis Using R b) Introduction to LaTeX	3	10	40	13 (25%)	2	2	1	5

Masters in Computer Science (Semester System)

**Scheme for
M.Sc. Computer Science (Semester IV)
Examination 2024
Session 2022-23**

Semester IV										
	Course Code	Course Title	Exam Hours	Max. Marks		Min. Marks	L	T	P*	Credits
				Int. Marks	Ext. Marks					
Core Courses										
1	FS-COMP- MSC-CS-CC- 401	Computer Graphics & Multimedia	3	10	40	13 (25%)	3	1	1	5
2	FS-COMP- MSC-CS-CC- 402	Android Programming	3	10	40	13 (25%)	3	1	1	5
Core Elective Courses										
3	FS-COMP- MSC-CS-CE- 403	a) Cloud Computing b) Internet of Things c) Big Data & Data Mining d) Machine Learning	3	10	40	13 (25%)	3	1	1	5
4	FS-COMP- MSC-CS-CP- 405	Combined Practical	6	25	75	26 (25%)	*combined practical of above subjects			
Open Elective Course										
1	FS-COMP- MSC-CS-EO- 406	a) Natural Language Processing b) Introduction to Cyber Security	3	10	40	13 (25%)	2	2	1	5

Masters in Computer Science (Semester System)

Learning Outcome Index

Learning Outcomes are statements of knowledge, skills, and abilities a student should possess and demonstrate upon completion of learning experiences.

I. Programme Outcomes(PO) and Programme Specific Outcomes (PSO)

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8	PSO9	PSO10	PSO11
PO1	X	X	X	X	X	X	X	X		X	X
PO2	X		X		X	X	X	X	X	X	X
PO3	X	X	X		X	X	X	X	X	X	X
PO4	X	X	X	X	X	X		X	X	X	X
PO5	X	X	X	X	X	X	X	X	X	X	X
PO6	X	X	X	X	X	X	X			X	X
PO7				X	X		X		X	X	X
PO8		X		X		X	X	X			X
PO9	X	X		X	X		X	X			X
PO10	X	X	X		X				X		X
PO11	X	X	X		X	X	X	X	X	X	X

II. Programme Specific Outcomes (PSO) and Core Courses (CC)

	MCS 101	MCS1 02	MCS 103	MCS 104	MCS 201	MCS 202	MCS 203	MCS 204	MCS 301	MCS 302	MCS 401	MCS 402
PSO1	X	X	X	X	X	X	X	X	X	X	X	X
PSO2	X	X	X	X	X	X	X	X	X	X	X	X
PSO3		X		X	X			X	X	X		X
PSO4	X	X	X	X	X	X	X	X	X	X	X	X
PSO5	X	X	X	X	X	X	X	X	X	X	X	X
PSO6	X		X		X	X	X				X	
PSO7	X	X	X	X	X	X	X	X	X	X	X	X
PSO8		X		X	X			X	X	X		X
PSO9		X	X	X	X			X	X	X	X	X
PSO10	X	X	X	X	X	X	X	X	X	X	X	X
PSO11	X	X	X	X	X	X	X	X	X	X	X	X

Masters in Computer Science (Semester System)

III. Programme Specific Outcomes (PSO) and Core Elective Courses (CEC)

	MCS 303a	MCS 303b	MCS 303c	MCS 303d	MCS 403a	MCS 403b	MCS 403c	MCS 403d
PSO1	X	X	X	X	X	X	X	X
PSO2	X	X	X	X	X	X	X	X
PSO3	X		X		X		X	
PSO4	X	X	X	X	X	X	X	X
PSO5	X	X	X	X	X	X	X	X
PSO6		X		X		X		X
PSO7	X	X	X	X	X	X	X	X
PSO8	X		X		X		X	
PSO9	X	X	X	X	X	X	X	X
PSO 10	X	X	X	X	X	X	X	X
PSO 11	X	X	X	X	X	X	X	X

IV. Programme Specific Outcomes (PSO) and Open Elective Courses (OEC)

	MCS 305a	MCS 305b	MCS 405a	MCS 405b
PSO1	X	X	X	X
PSO2	X	X	X	X
PSO3	X		X	
PSO4	X	X	X	X
PSO5	X	X	X	X
PSO6		X		X
PSO7	X	X	X	X
PSO8	X		X	
PSO9	X	X	X	X
PSO 10	X	X	X	X
PSO 11	X	X	X	X

Masters in Computer Science (Semester System)

Objectives, Course-level Learning Outcomes, Contents, and Suggested Readings

Semester I

Paper Code:FS-COMP-MS-C-CC-101

Paper Name: Mathematics for Computer Science

Course Objectives:

- CO1. To learn to evaluate mathematical arguments revolving around computation
- CO2. To understand the basics of Combinations and Permutations
- CO3. To acquire the ability to represent relations matrices and digraphs
- CO4. To acquire and apply the knowledge on Graphs and Trees to real-world applications
- CO5. To have the ability to Demonstrate the working of Grammars and Languages

Learning Outcomes:

After completion of this course, the student will be able to-

- LO1. Comprehend and evaluate mathematical arguments revolving around computation.
 - LO2. Understand the basics of Combinations and Permutations.
 - LO3. Represent relations matrices and digraphs.
 - LO4. Apply the knowledge on Graphs and Trees to real-world applications.
 - LO5. Demonstrate the working of Grammars and Languages.
-

Note: Non-Scientific Calculator may be allowed in the end-semester examination.

Course Description

Unit – I

Sets: different types of sets, set operations; Basic Counting Principles, Pigeonhole Principle, Binomial Coefficients, Binomial Theorem, Permutations, Combinations; **Matrices:** addition, multiplication; **Vectors:** position vector, addition, subtraction and products of vectors.

Unit - II

Mathematical Induction; **Logic:** Propositions and logical operations, Conditional statements, Tautologies and Contradictions, Logical Equivalence, quantifiers. Basic computability theory: Chomsky Hierarchy, the concept of models of computation, the concept of types of languages and grammars.

Unit - III

Relations: Representation of Relations, Properties of relations, transitive closure; Ordered Sets: poset, Properties, Hasse Diagram, Extremal elements of posets; **Functions:** Types of Functions, Asymptotic notations; Coordinate Systems: representation of points, straight lines, standard equation of circles.

Recommended Readings

Masters in Computer Science (Semester System)

1. K.H. Rosen, Discrete Mathematics and its applications, seventh edition
2. Kolman, Busby and Ross, Discrete Mathematical Structures, Sixth Edition, PHI.

Suggested Readings

3. Schaum's Outline Of Theory and Problems of Discrete Mathematics, Third Edition.
4. C L Liu, Elements of Discrete Mathematics, TMH,
5. John Vince, Foundation Mathematics for Computer Science: A Visual Approach, Springer
6. George B. Thomas and Ross L. Finney, Calculus and Analytic Geometry, Addison Wesley
7. J. Ullman and J. Hopcroft , Introduction to Automata Theory, Languages, and Computation, Pearson Education
8. Daniel I.A. Cohen, Introduction to Computer Theory, 2ed, Wiley.
9. Peter Linz, An Introduction to Formal Languages and Automata, Sixth edition.

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MS-C-CC-102

Paper Name: Internet Programming

Course Objectives -

CO1. To gain knowledge of how the client-server model of Internet programming works

CO2. To learn design and development of interactive, client-side, executable web applications

CO3. To acquire the ability to demonstrate how Internet programming tasks are accomplished

CO3: To know how to build tools that assist in automating data transfer over the Internet.

CO4: To understand the advantages and disadvantages of the core Internet protocols

Learning Outcomes:

After completion of this course, the student will be able to-

LO1: Explain how the client-server model of Internet programming works

LO2: Design and develop interactive, client-side, executable web applications

LO3: Demonstrate how Internet programming tasks are accomplished

LO3: Build tools that assist in automating data transfer over the Internet

LO4: Compare the advantages and disadvantages of the core Internet protocols

Course Description

Unit I

Internet Basics: Evolution of the Internet, Basic internet terms and applications. ISP, Anatomy of an e-mail Message, basics of sending and receiving, E-mail Protocol; Mailing List- Subscribing, Unsubscribing. Introduction to World Wide Web and its work, Web Browsers, Search Engine, Downloading, HyperText Transfer Protocol (HTTP), URL, Web Servers, FTP, Web publishing- Domain Name Registration, Space on Host Server for Web Site, Maintain and Updating.

Unit - II

HTML: Elements of HTML & Syntax, Comments, Headings, Paragraph, Span, Pre Tags, Backgrounds, Formatting tags, Images, Hyperlinks, div tag, List Type and its Tags, Table Layout, Use of Forms in Web Pages. CSS: Introduction to Cascading Style Sheets, Types of Style Sheets (Inline, Internal and External), using Id and classes, CSS properties: Background Properties, Box Model Properties, Margin, Padding, List Properties, Border Properties, Positioning Properties.

Unit - III

Java Script: Introduction to Client-Side Scripting, Introduction to JavaScript, Comments, Variables in JS, Global Variables, Data types, Operators in JS, Conditions Statements (If, If Else, Switch), JavaScript Loops (For Loop, While Loop, Do While Loop), JS Popup Boxes (Alert, Prompt, Confirm), JS Events, Onload, Onunload, Onsubmit, Onfocus, Onchange Event, Onblur Event, Onmouseover, Onclick, Ondblclick Events, JS Arrays, Working with Arrays, JS Objects, Window object, Document object, JS Functions, getElementById, innerHTML property, inner Text property, form validation, email validation.

Recommended Readings

1. Thomas A. Powell, HTML: The Complete Reference, Osborne/McGraw-Hill

Masters in Computer Science (Semester System)

2. Deitel, Deitel and Nieto, Internet & WWW. How to program, 2nd Edition, Pearson Education Asia.

Suggested Readings

3. E Stephen Mack, Janan Platt, HTML 4.0, No Experience Required, 1998, BPB Publications.
4. Sybex, HTML Complete, BPB Publications.
5. V.K Jain, Internet and Web Page Designing, BPB Publications.
6. Ivan Bayross, Web Enabled Commercial Application Development Using HTML, DHTML, java script, Perl CGI, BPB publications.

Paper Code:FS-COMP-MS-C-CC-103

Paper Name: Computer Organization

Course Objectives:

CO1: To understand the structure, function, and characteristics of computer systems.

CO2: To understand the design of the various functional units and components of computers.

Masters in Computer Science (Semester System)

CO3: To Identify the elements of modern instruction sets and their impact on processor design.

CO4: To acquire the ability to explain the function of each element of a memory hierarchy,

CO5: To identify and compare different methods for computer I/O

Learning Outcomes:

After completion of this course, the student will be able to-

LO1: Understand the structure, function, and characteristics of computer systems.

LO2: Understand the design of the various functional units and components of computers.

LO3: Identify the elements of modern instruction sets and their impact on processor design.

LO4: Explain the function of each element of a memory hierarchy,

LO5: Identify and compare different methods for computer I/O.

Unit I

Components of a Computer: Processor, Memory, Input-Output Unit, Difference between Organization and Architecture, Hardware-Software Interaction. **Number System:** Concept of Bit and Byte, types, and conversion. **Complements:** 1's complement, 2's complement. **Binary Arithmetic:** Addition, overflow, subtraction, multiplication (booth's algorithm), and division algorithm. **Logic gates:** Boolean Algebra, Map Simplification.

Unit II

Combinational circuits: Half Adder, Full Adder, Decoders, Multiplexers. **Sequential circuits:** Flip Flops- SR, JK, D, T Flip-Flop, Excitation Tables, State Diagram, State Table, Registers, Counters.

Input-Output Organization: Peripheral devices, I/O Interface, Asynchronous Data Transfer, Modes of Data Transfer, Priority Interrupt, Direct Memory Access, I/O Processor.

Memory Organization: Types and capacity of Memory, Memory Hierarchy, Associative Memory, Buffer, Cache Memory, Virtual Memory.

Unit III

Intel 8085 Microprocessor: Introduction, ALU, Timing and Control Unit, Register Set, Data and Address Bus, Addressing modes, Complete Intel 8085 Instruction set, Instruction format, Opcode and Operand, Word Size, Instruction Cycle, Pin Configuration, Intel 8085 programs.

Recommended Readings

1. M. Morris Mano, Computer System Architecture, Pearson, Prentice Hall.
2. J.P. Hayes, Computer Architecture & Organization, Tata McGraw Hill

Suggested Readings

3. Malvino Leach and Jerald A. Brown, Digital Computer Electronics, McGraw Hill.
4. Ramesh Gaonkar, Microprocessor Architecture, Programming, and Application With the 8085, PENRAM.

Masters in Computer Science (Semester System)

5. B.Ram, Fundamentals of Microprocessor and Microcomputers, Danpat Rai Publications.

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MS-C-CC-104

Paper Name : C++ Programming

Course Objectives:

- CO1. To declare, initialize and process variables, constants, and arrays
- CO2. To read and print values from the keyboard using Scanner and Dialog boxes
- CO3. To create statements for decisions and loops
- CO4. To define functions and return values
- CO5. To create classes, objects, and constructors
- CO6. To understand and apply OO design concepts
- CO7. To create, open, manipulate and close files using Streams
- CO8. To create applets for drawing shapes and playing audio clips

Learning Outcomes:

After completion of this course, the student will be able to-

- LO1. Declare, initialize and process variables, constants, and arrays
 - LO2. Read and print values from the keyboard using Scanner and Dialog boxes
 - LO3. Create statements for decisions and loops
 - LO4. Define functions and return values.
 - LO5. Create classes, objects, and constructors.
 - LO6. Understand and apply OO design concepts.
 - LO7. Create, open, manipulate and close files using Streams.
 - LO8. Create applets for drawing shapes and playing audio clips.
-

Unit I

Object-Oriented System Object-Oriented Paradigm: need, characteristics, applications. Basics of C++, branching, looping, and jump statements. **Functions:** need, types, passing arguments by value and reference, recursive function, pointers, and functions. **Arrays:** need, types, array and function, array and pointers.

Unit II

Class: Basics, static data members, Inline Function, Constructors and Destructors: need, types, usage, **Inheritance** - need, usage, types, compile-time and run-time polymorphism, overloading and overriding, virtual function, friend function, abstract class. **Operator overloading:** need, rules, through member function and through friend function.

Unit III

String handling, String class, Templates, Additional Features for C++ 11, C++14 and C++17 Searching and Sorting: **Searching:** Linear Search, Binary Search. **Sorting:** Insertion Sort, Selection Sort, Quick Sort, Bubble Sort, Heap Sort, Shell Sort, Merge sort, Radix Sort, Counting Sort, Bucket Sort.

Recommended Readings

Masters in Computer Science (Semester System)

1. E. Balagurusamy, Object-Oriented Programming With C++ , Tata Mcgraw Hill.
2. Herbert Schildt, C++ The Complete Reference, Tata Mcgraw Hill.

Suggested Readings

3. Schaum Series, Object Oriented Programming With C++, Tata Mcgraw Hill.
4. Paul J. Deitel and Harvey M. Deitel, C++11 for Programmers (Deitel Developer), Prentice Hall; 2nd edition.
5. Marc Gregoire, Nicholas A. Solter and Scott J.Kleper, Professional C++, Goodreads Publications.
6. Bjarne Stroustrup, A Tour of C++.
7. Bartłomiej Filipek, C++17 in Detail.

Masters in Computer Science (Semester System)

Semester II

Paper Code:FS-COMP-MSC-CS-CC-201

Paper Name: Database Management System

Course Objectives:

- CO1: To understand the need for a DB approach and understand the components and roles of DBMS
- CO2: To know how to write SQL queries for the given problem statement
- CO3: To apply DB system development life cycle to business problems
- CO4: To develop ER diagram for representing the conceptual data model
- CO5: To convert ER diagram into a set of relations representing the logical data model
- CO6: To implement a collection of ties in the chosen DBMS product, such as ORACLE
- CO7: To have a broad understanding of database concepts and database management system software
- CO8: To have a high-level experience of major DBMS components and their function
- CO9: To be able to model an application's data requirements using conceptual modeling tools like ER diagrams and design database schemas based on the conceptual model.
- CO10: To be able to write SQL commands to create tables and indexes, insert/update/delete data, and query data in a relational DBMS.
- CO11: To understand detailed architecture, define objects, load data, query data, and performance tune SQL databases.
- CO12: To be able to handle large volumes of structured, semi-structured, and unstructured data using database technologies.

Learning Outcomes:

After completion of this course, the student will be able to-

- LO1: Appreciate the need for a DB approach and understand the components and roles of DBMS
- LO2: Write SQL queries for the given problem statement
- LO3: Apply DB system development life cycle to business problems
- LO4: Develop ER diagram for representing the conceptual data model
- LO5: Convert ER diagram into a set of relations representing the logical data model
- LO6: Implement a collection of ties in the chosen DBMS product, such as ORACLE
- LO7: Have a broad understanding of database concepts and database management system software
- LO8: have a high-level experience of major DBMS components and their function
- LO9: be able to model an application's data requirements using conceptual modeling tools like ER diagrams and design database schemas based on the conceptual model.
- LO10: be able to write SQL commands to create tables and indexes, insert/update/delete data, and query data in a relational DBMS.
- LO11: To understand detailed architecture, define objects, load data, query data, and performance tune SQL databases.

Masters in Computer Science (Semester System)

LO12: Able to handle large volumes of structured, semi-structured, and unstructured data using database technologies.

Unit I

Introduction: Characteristics of database approach, Advantages, Database system architecture, Overview of different types of Data Models and data independence, Schemas and instances, Database languages and interfaces; **E-R Model:** Entities, Attributes, keys, Relationships, Roles, Dependencies, E-R Diagram; Normalization: Definition, Functional dependencies and inference rules, 1NF, 2NF, 3NF, and BCNF.

Unit II

Introduction to Relational model, Constraints: Domain, Key, Entity integrity, Referential integrity; Keys: Primary, Super, Candidate, Foreign; **Relational algebra:** select, project, union, intersection, minus, cross product, different types of join, division operations; aggregate functions and grouping; **SQL:** Data Types, statements: select, insert, update, delete, create, alter, drop; views, SQL algebraic operations, nested queries; Stored procedures: Advantages, Variables, creating and calling procedures, if and case statements, loops, Cursors, Functions, Triggers.

Unit III

Transactions processing: Definition, desirable properties of transactions, serial and non-serial schedules, the concept of serializability, conflict-serializable schedules; **Concurrency Control:** Two-phase locking techniques, dealing with Deadlock and starvation, deadlock prevention protocols, basic timestamp ordering algorithm; Overview of database recovery techniques; the concept of data warehousing.

Recommended Readings

1. Ramez A. Elmasri, Shamkant Navathe, Fundamentals of Database Systems, 5th Ed, Pearson Publications.
2. Korth, Silberschatz, Sudarshan, Database System Concepts, Mcgraw Hill.

Suggested Readings

3. Bipin C. Desai, An Introduction to Database Systems, Galgotia Publication.
4. Ivan Bayross, SQL, PL/SQL Programming, BPB publications.
5. Ivan Bayross, Commercial Application Development Using Oracle Developer 2000, BPB publications.

Web Resources

1. <http://www.mysqltutorial.org/mysql-stored-procedure-tutorial.aspx>

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MS-C-CC-202

Paper Name: Data Communication and Networking

Course

Objectives:

After completion of this course the student will be able to-

- CO1. To gain the ability to create a new protocol and test its efficiency
- CO2. To design a new network architecture using protocols and interfaces
- CO3. To create a hybrid topology using the existing topologies, and check inefficiency
- CO4. To apply different encoding and decoding mechanisms involved in various types of transmission media and measure the transmission impairments
- CO5. To design a model internet with various categories of networks and test the transmission rate
- CO6. To understand the basics of data communication, networking, the internet, and their importance
- CO7. To analyze the services and features of various protocol layers in data networks
- CO8. To differentiate wired and wireless computer networks
- CO9. To analyze TCP/IP and their protocols
- CO10. To recognize the different internet devices and their functions
- CO11. To identify the primary security threats of a network

Learning Outcomes:

After completion of this course the student will be able to-

- LO1. Create a new protocol and test its efficiency.
- LO2. Design a new network architecture using protocols and interfaces.
- LO3. Create a hybrid topology using the existing topologies, and check inefficiency.
- LO4. Apply different encoding and decoding mechanisms involved in various types of transmission media and measure the transmission impairments.
- LO5. Design a model internet with various categories of networks and test the transmission rate.
- LO6. Understand the basics of data communication, networking, the internet, and their importance.
- LO7. Analyze the services and features of various protocol layers in data networks.
- LO8. Differentiate wired and wireless computer networks.
- LO9. Analyze TCP/IP and their protocols.
- LO10. Recognize the different internet devices and their functions.
- LO11. Identify the primary security threats of a network.

Unit - I

Data Communication and Networking: Overview, Network Types, LAN Technologies, Topologies, Models- OSI Model, TCP/IP Stack, Security

Physical Layer: Introduction, Impairments, Performance, Digital Transmission, modes, digital to digital, analog to digital, Analog Transmission, digital to analog, analog to analog, Transmission media, Wireless Transmission, Multiplexing, FDM, TDM, CDM, WDM,

Switching techniques: Circuit Switching, Packet switching, Datagram, Virtual circuit, and Permanent Virtual Circuit, Connectionless and connection-oriented communication, Message switching,

Masters in Computer Science (Semester System)

Unit - II

Data Link Layer: Introduction, Error Detection, and Correction, Data Link Control: Line Discipline- Enq/Ack, Poll/Select, **Flow Control:** Stop And Wait, Sliding Window, **Error Control:** ARQ, Stop and Wait ARQ, Sliding Window ARQ.

Network Layer: Introduction, Network Addressing, Routing, Internetworking, Tunneling, Packet Fragmentation, Network Layer Protocols, ARP, ICMP, IPv4, IPv6

Transport Layer: Introduction, Function, End to end communication, Transmission Control Protocol, User Datagram Protocol

Application Layer: Introduction, Client-Server Model, Application Protocols, Network Services

Unit - III

Cyber Security: definition, cybercrime and information security, cybercriminals, classification of cybercrime. Cyber offenses: categories of cybercrime.

Tools and methods used in cybercrime: phishing, types of phishing, types, and techniques of ID theft, password cracking, keyloggers and spyware, backdoors, steganography, DoS, SQL Injection.

Cybercrime on mobile and wireless devices: attacks on wireless networks, Authentication security service, attacks on mobile phones. Cyber Law, The Indian IT Act, Digital Signatures, Anti- Cybercrime Strategies, Cyberterrorism, Indian ITA 2000.

Recommended Readings

1. Nina Godbole & Sunit Belapur, Cyber Security.
2. Forozan, Data Communication and Networking, Tata McGraw Hill.

Suggested Readings

3. Dr. Madhulika Jain, Satish Jain, Data Communication And Computer Networks, BPB publications.
4. William Stallings, Data and Computer Communications, Pearson Education.
5. A. S. Tanenbaum, Computer Networks, Fourth Edition, Pearson Education.

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MS-C-CC-203

Paper Name: Operating System

Course Objectives:

- CO1. To be able to design and understand the following OS components: System calls, Schedulers, Memory management systems, Virtual Memory, and Paging systems.
- CO2. To be able to evaluate, and compare OS components through instrumentation for performance analysis.
- CO3. To analyze the various device and resource management techniques for time-sharing and distributed systems
- CO4. To develop and analyze simple concurrent programs using transactional memory and message passing, and understand the trade-offs and implementation decisions

Learning Outcome:

After completion of this course, the student will be able to-

- LO1. Allocate Main Memory based on various memory management techniques
 - LO2. Compare Memory allocation using Best fit, Worst fit, and first hold policies
 - LO3. Apply page replacement policies for dynamic memory management
 - LO4. Schedule CPU time using scheduling algorithm for processors
 - LO5. Compare various device scheduling algorithms. serve
-

Unit I

Introduction to Operating System, layered Structure, Functions, Types; Process: Concept, Process States, PCB; Threads, System calls; Process Scheduling: types of schedulers, context switch, CPU Scheduling, Preemptive Scheduling, Scheduling Criteria- CPU Utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling Algorithms- FCFS, SJF, Priority Scheduling, Round Robin Scheduling, MLQ Scheduling, MLQ With Feedback.

Unit II

Synchronization: Critical Section Problem, Requirements for a solution to the critical section problem; Semaphores, simple solution to Readers-Writers Problem. Deadlock: Characterization, Prevention, Avoidance, Banker's Algorithm, Recovery from Deadlock. Memory Management: Physical and virtual address space, Paging, Overview of Segmentation; Virtual Memory Management: Concept, Page Replacement techniques- FIFO, LRU, Optimal

Unit III

Linux: features of Linux, steps of Installation, Shell and kernel, Directory structure, Users and groups, file permissions, commands- ls, cat, cd, pwd, chmod, mkdir, rm, rmdir, mv, cp, man, apt, cal, uname, history etc.; Installing packages; Shell scripts: writing and executing a shell script, shell variables, read and expr, decision making (if-else, case), for and while loops.

Recommended Readings

1. Abraham Silberschatz, Peter Baer Galvin, Operating System Principles, John Wiley And Sons Inc.
2. Milan Milen Kovic, Operating System Concepts And Design, Tata Mcgraw Hill.

Masters in Computer Science (Semester System)

Suggested Readings

3. Andrew S. Tanenbaum, Herbert Bos, Modern Operating System.
4. Mike McGrath, Linux in easy steps.
5. Sumitabha Das, Unix concepts and applications, TMH.

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MSC-CS-CC-204

Paper Name: PHP

Unit – I

PHP: Installation of PHP. **Building Blocks of PHP:** Variables, data types, Operators & Expressions, Constants, Switching, Flow, Loops. **Functions:** Meaning, Calling, Defining a function. Return value from user defined function. **Arrays:** Creating arrays, Array related functions. **Working with String, Date & Time:** Formatting String with PHP, Using Date and time Functions with PHP. Working with file and Directories.

Unit - II

Forms: Creating simple input Form. Accessing Form input with user defined arrays, HTML and PHP Code on a single page. Redirecting User. Working with File Upload. Uploading & Downloading.

State management: Using query string(URL rewriting), Using Hidden field, Using cookies, Using session.

Unit - III

Exception Handling: Understanding Exception and error, Try, catch, throw

Connecting to the MYSQL: Selecting a database, Adding data to a table, Displaying returned data on Web pages, Inserting data, Deleting data, Entering and updating data, Executing multiple queries, executing stored procedures.

Suggested Readings:

1. Teach Yourself PHP, MYSQL & Apache By Meloni, Pearson Education.
2. Open Source Development with LAMP: Using Linux, Apache, MySQL, Perl & PHP By James Lee, Pearson Education.
3. PHP: A Beginner's Guide By Vaswani, Vikram Tata Mc-Graw Hill.

Masters in Computer Science (Semester System)

Semester III

Paper Code:FS-COMP-MS-C-CC-301

Paper Name: Data Structures

Course Objectives:

- CO1. To Create and initialize variables, constants, arrays, pointers, structures, and unions.
- CO2. To Manipulate values of variables, arrays, pointers, structures, unions, and files.
- CO3. To create a function that can receive variables, arrays, pointers, and structures.
- CO4. To define functions that can receive variables, arrays, pointers, and structures.
- CO5. To create open, read, manipulate, write and close files.
- CO6. To select and use appropriate data structures for the given problems.
- CO7. To design efficient algorithms using various algorithm designing strategies
- CO8. To analyze the problem and develop the algorithms related to these problems
- CO9. To classify the problem and apply the appropriate design strategy to develop an algorithm
- CO10. To design algorithm in the context of space and time complexity and apply the asymptotic notation
- CO11. To be able to analyze algorithms and algorithm correctness.
- CO12. To be able to summarize searching and sorting techniques
- CO13. To be able to describe stack, queue, and linked list operations.
- CO14. To be able to know. tree and graphs concepts

Learning Outcomes:

After completing this course, students will be able to:

- LO1. Create and initialize variables, constants, arrays, pointers, structures, and unions.
 - LO2. Manipulate values of variables, arrays, pointers, structures, unions, and files.
 - LO3. Create a function that can receive variables, arrays, pointers, and structures.
 - LO4. Define functions that can receive variables, arrays, pointers, and structures.
 - LO5. Create open, read, manipulate, write and close files.
 - LO6. Select and use appropriate data structures for the given problems.
 - LO7. Design efficient algorithms using various algorithm designing strategies
 - LO8. Analyze the problem and develop the algorithms related to these problems
 - LO9. Classify the problem and apply the appropriate design strategy to develop an algorithm
 - LO10. Design algorithm in the context of space and time complexity and apply the asymptotic notation
 - LO11. Ability to analyze algorithms and algorithm correctness.
 - LO12. Ability to summarize searching and sorting techniques
 - LO13. Ability to describe stack, queue, and linked list operations.
 - LO14. Ability to know. tree and graphs concepts
-

Unit I

Algorithm: Efficiency & Analysis Algorithm: Time and Space complexity of Algorithm.

Abstract Data Type: Linked List- Linear, Circular, Two Way List, Basic Operation on Linked Lists, Application of Linked List.

Unit II

Stack: primitive operations, stack Application- Infix, postfix, prefix and Recursion Array, and Linked Representation of Stack. **Queue:** Primitive operation, Circular Queue, Priority Queue, D-queue, Array, and Linked Representation of Queue.

Masters in Computer Science (Semester System)

Unit III

Trees: Basic terminology, **Binary Tree:** Representation as Array and link List, Basic operation, **Tree Traversal:** Inorder, Preorder, Postorder, Application of Binary Tree. B-tree, Height Balance Tree (AVL Tree) **Graph:** Basic Terminology, Directed, Undirected, Weighted, Representation of Graphs, **Graph Traversal:** Depth First Traversal, Breadth-First Search.

Recommended Readings

1. R.B Patel, Expert Data Structure with 'C', Khana Book Publishing.
2. Lipschutz, Data structure, Tata McGraw Hill.

Suggested Readings

3. Yashvant Kanitkar, Data Structure, BPB publications.
4. Jean-Paul Tremblay, Paul G.Sarerson, An Introduction to Data Structures with Applications, Tata McGraw Hill.
5. Yedidyah Langsam, Moshe J.Augenstein, Arora M. Tenenbaum, Data Structure Using C and C++, Prentice- Hall India

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MS-C-CC-302

Paper Name: Java

Course Objectives:

CO1. To use an integrated development environment to write, compile, run, and test simple object-oriented Java programs.

CO2. To read and make elementary modifications to Java programs that solve real-world problems.

CO3. To validate input in a Java program.

CO4. To identify and fix defects and common security issues in code.

CO5. To document a Java program using Javadoc.

CO6. To use a version control system to track source code in a project.

Learning Outcomes:

After completing this course, students will be able to:

LO1. Use an integrated development environment to write, compile, run, and test simple object-oriented Java programs.

LO2. Read and make elementary modifications to Java programs that solve real-world problems.

LO3. Validate input in a Java program.

LO4. Identify and fix defects and common security issues in code.

LO5. Document a Java program using Javadoc.

LO6. Use a version control system to track source code in a project.

Unit I

Introduction to Java: evolution, features, comparison with C and C++; Java program structure; tokens, keywords, constants, variables, data types, typecasting, statements, Operators and Expression; Conditional Statements and Loop Statements. **Class:** syntax, instance variable, class variables, methods, constructors, overloading.

Unit II

Inheritance: types of inheritance, use of super, method overriding, final class, abstract class, wrapper classes.

Arrays, Strings and Vectors, Packages and Interfaces, visibility controls

Unit III

Errors and Exceptions: Types of errors, Exception classes, Exception handling in java, use of try, catch, finally, throw and throws. Taking user input, Command line arguments.

Multithreaded Programming: Creating Threads, the Life cycle of thread, Thread priority, Thread synchronization, Inter-thread communication, Implementing the Runnable Interface.

Recommended Readings

1. Herbert Schildt, The Complete reference Java Ninth Edition, Tata McGraw Hill
2. Burd, Beginning Programming with Java For Dummies , For Dummies; 3 edition

Suggested Readings

3. Herbert Schildt, Java: A Beginner's Guide, Sixth Edition: A Beginner's Guide, McGraw-Hill
4. E. Balagurusamy, Osborne MediaProgramming in JAVA, TMH.
5. Steven Holzner et al. JAVA 2 programming Black Book, Dreamtech Press.
6. E. Balagurusamy, Programming in JAVA, TMH.

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MS-CE-303(a)

Paper Name: Software Engineering & Research Methodology

Course Objectives:

- CO1. To learn the phases of software development
- CO2. To develop process models and process systems multiple collections, models
- CO3. To gather, understand, analyze and specify requirements
- CO4. To develop architectural diagram, and implement by following coding principles
- CO5. To apply testing strategies and handle software product maintenance issues
- CO6. To get a good knowledge of the issues and challenges faced while doing the Software project Management.
- CO7. To understand why the majority of the software projects fail and how that failure probability can be reduced effectively.
- CO8. To do the Project Scheduling, tracking, Risk analysis, Quality management, and Project Cost estimation using different techniques.
- CO9. To identify and discuss the role and importance of research in the social sciences.
- CO10. To identify and discuss the issues and concepts salient to the research process.
- CO11. To identify and discuss the complex issues inherent in selecting a research problem, selecting an appropriate research design, and implementing a research project.
- CO12. To identify and discuss the concepts and procedures of sampling, data collection, analysis, and reporting.

Learning Outcomes:

After completing this course, students will be able to:

- LO1. Learn the phases of software development
 - LO2. Develop process models and process systems multiple collections, models
 - LO3. Gather, understand, analyze and specify requirements
 - LO4. Develop architectural diagram, and implement by following coding principles
 - LO5. Apply testing strategies and handle software product maintenance issues
 - LO6. Get a good knowledge of the issues and challenges faced while doing Software project Management.
 - LO7. To understand why the majority of the software projects fail and how that failure probability can be reduced effectively.
 - LO8. To do the Project Scheduling, tracking, Risk analysis, Quality management, and Project Cost estimation using different techniques.
 - LO9. Identify and discuss the role and importance of research in the social sciences.
 - LO10. Identify and discuss the issues and concepts salient to the research process.
 - LO11. Identify and discuss the complex issues inherent in selecting a research problem, selecting an appropriate research design, and implementing a research project.
 - LO12. identify and discuss the concepts and procedures of sampling, data collection, analysis, and reporting.
-

Unit I

Software: Software Characteristics, Software Process, Process Characteristics, **Software Process Model:** Linear Sequential Model, Prototyping Model, Spiral Model, Software Quality, McCall's Quality Factors, **Software Requirement Analysis and Specification (SRS):** Need Characteristics and Components.

Unit II

Planning a Software Project: COCOMO Model, Project Monitoring Plan, and Risk Management. **Design Principle:** Abstraction, Modularity, Cohesion and Coupling, **Software Management:** Size Oriented Metrics, Function Oriented Metrics. **Testing:** Testing

Masters in Computer Science (Semester System)

Fundamental, Functional Testing (Black Box), Structural Testing (White Box), Alpha And Beta Testing, **Testing Process:** Comparison of Different Testing, Level of Testing.

Unit III

Research Methodology: Meaning of Research, Objective of Research, Types of Research, Research Approaches, Significance of research, Research Methods versus Methodology, Research Process, Criteria of Good Research, What is Research Problem, Selecting the problem, Necessity of defining the problem, Technique involved in defining a problem.

Recommended Readings

1. Roger S. Pressman, Software Engineering: A Practitioner's Approach, McGraw Hill.

Suggested Readings

2. Pankaj Jalote, Software Engineering: A Precise Approach, Wiley Precise textbook Series.
3. C. R. Kothari, Research Methodology Methods and Techniques, New Age International Publisher.

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MS-C-CE-303(b)

Paper Name: Artificial Intelligence

Course Objectives:

CO1. To analyze and formalize the problem as a state space, graph, design heuristics

CO2. To have the ability to represent solutions for various real-life problem domains using logic-based techniques

CO3. To understand the numerous applications and huge possibilities in the field of AI

CO4. To ability to express ideas in AI research and programming language related to emerging technology.

Learning Outcomes:

After completing this course, students will be able to:

LO1. To analyze and formalize the problem as a state space, graph, design heuristics

LO2. Ability to represent solutions for various real-life problem domains using logic-based techniques

LO3. Understand the numerous applications and huge possibilities in the field of AI

LO4. Ability to express ideas in AI research and programming language related to emerging technology.

Unit I

Definition, History, Agents, and environment, Defining the problem as a state and space search, What is Intelligence? Types of Intelligence, Difference between Human and Machine Intelligence, The Structure of Intelligent Agents. Solving problems by searching: Uninformed search strategies- Brute-Force, Breadth-First, Uniform-cost search Depth-First, Depth-limited search,depth-first search, Bidirectional search. Informed (heuristic) search strategies- Greedy best-first search, A*, AO* Memory-bounded heuristic search.

Unit II

Heuristic functions, local search algorithms- Hill-climbing search, Simulated annealing, Local beam search. Knowledge-Based System: Knowledge, Procedure V/S Declarative Knowledge, Knowledge Representation: Using Procedural and Predicate Logic, Inference in First-order logic: Unification and Lifting, Forward Chaining, Backward Chaining, Resolution. Rule-based System, Frames, Frames, Scripts, and Semantic Nets.

Unit III

Probabilistic Reasoning, Probability, and Bayes Theorem represent knowledge in the uncertain domain, Certainty factors, Bayesian Networks, Dempster-Shafer theory, introduction to Fuzzy logic. Learning: types of learning, decision trees. **Expert System: types, architecture. Introduction to Artificial Neural Networks, Reinforcement Learning, Natural Language Processing, Pattern Recognition, and Perception.**

Recommended Readings

1. Rich And Knight, Artificial Intelligence, Tata McGraw Hill

Suggested Readings

2. Patterson, Introduction to Artificial Intelligence and Expert Systems, Prentice-Hall India.
3. Russell and Norvig, Artificial Intelligence A Modern Approach, Prentice Hall.

Masters in Computer Science (Semester System)

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MSC-CS-CE-303(c)

Paper Name: Python

Course Objectives:

- CO1. Apply language features including strings, lists, tuples, dictionaries, regular expressions.
- CO2. Create and call functions.
- CO3. Create and manipulate files.
- CO4. Develop classes using OO features.
- CO5. Develop internet applications using packages such as urllib.
- CO6. To understand why Python is a proper scripting language for developers.
- CO7. To learn how to design and program Python applications.
- CO8. To learn how to use lists, tuples, and dictionaries in Python programs.
- CO9. To learn how to identify Python object types.
- CO10. To learn how to use indexing and slicing to access data in Python programs.
- CO11. To define the structure and components of a Python program.
- CO12. To learn how to write loops and decision statements in Python.
- CO13. To learn how to write functions and pass arguments in Python.
- CO14. To learn how to build and package Python modules for reusability.
- CO15. To learn how to read and write files in Python.
- CO16. To learn how to design object-oriented programs with Python classes.
- CO17. To learn how to use class inheritance in Python for reusability.
- CO18. To learn how to use exception handling in Python applications for error handling.

Learning Outcomes:

After completing this course, students will be able to:

- LO1. Apply language features including strings, lists, tuples, dictionaries, regular expressions. LO2. Create and call functions.
- LO3. Create and manipulate files.
- LO4. Develop classes using OO features.
- LO5. Develop internet applications using packages such as urllib.
- LO6. To understand why Python is a proper scripting language for developers.
- LO7. To learn how to design and program Python applications.
- LO8. To learn how to use lists, tuples, and dictionaries in Python programs.
- LO9. To learn how to identify Python object types.
- LO10. To learn how to use indexing and slicing to access data in Python programs.
- LO11. To define the structure and components of a Python program.
- LO12. To learn how to write loops and decision statements in Python.
- LO13. To learn how to write functions and pass arguments in Python.
- LO14. To learn how to build and package Python modules for reusability.
- LO15. To learn how to read and write files in Python.
- LO16. To learn how to design object-oriented programs with Python classes.
- LO17. To learn how to use class inheritance in Python for reusability.
- LO18. To learn how to use exception handling in Python applications for error handling.

Unit I

Basics: Python Interpreter, writing code in Jupyter Notebook, Indentation, comments, importing a module, binary operators, standard scalar data types, typecasting, if-else statements, loops(while, for), pass, range, ternary expressions. Data Structures and Sequences: Tuples, Lists, and slicing, Built-in Sequence functions, Dictionary, Sets; List, Set, and Dict Comprehensions.

Masters in Computer Science (Semester System)

Unit II

Functions: Namespaces, Scope, and Local Functions; Returning Multiple Values, Anonymous (Lambda) Functions, Partial Argument Application, Generators, Errors, and Exception handling. Basic File Handling. Objects and Methods in Python. NumPy: creating N-dimensional arrays, arithmetic with NumPy arrays, basic indexing, and slicing, Psuedorandom number generation.

Unit III

Pandas: Overview of Series and DataFrames, reading data from csv file, DataFrame operations- working with data using functions like head, tail , info, shape, reshape, columns, isnull, dropna, mean, sum, describe, value_counts, corr, loc, iloc, apply. Matplotlib- plotting basic figures, subplots, line plots, bar plots, histograms, scatter plots. Overview of Scikit-learn, SciPy, networkx. Applications of python.

Recommended Readings

1. Wes McKinney, Python for Data Analysis: Data Wrangling with Pandas, NumPy, and Ipython, O'Reilly Media.
2. John Shovic and Alan Simpson, Python All-in-One for Dummies, John Wiley & Sons, Inc.

Suggested Readings

3. Mark Summerfield, Programming in Python 3: A Complete Introduction to the Python Language, Pearson.
4. Swaroop, C. H. A Byte of Python. Python Tutorial.
5. John V. Guttag, Introduction to Computation and Programming Using Python, MIT Press.
6. Mark Lutz, David Ascher, Python, O'Reilly.
7. T. Budd, Exploring Python, TMH.

Web Resources

1. <https://www.learnpython.org/>
2. <https://nptel.ac.in/courses/106/106/106106212/>
3. <http://greenteapress.com/thinkpython/thinkpython.pdf>
4. Python tutorial: <https://docs.python.org/3/tutorial/index.html>
8. Python All-in-One for Dummies, by John Shovic and Alan Simpson, John Wiley & Sons, Inc., 2019

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MS-C-CE-303(d)

Paper Name: Theory of Computation

Course Objectives:

- CO1. able to design Finite Automata machines for given problems;
- CO2. able to analyze a given Finite Automata machine and find out its Language;
- CO3. able to create Pushdown Automata machine for given CF language(s);
- CO4. able to generate the strings/sentences of given context-free languages using its grammar;
- CO5. Able to design Turing machines for given Apply to identify Interpretational problem.

Learning Outcomes:

After completing this course, students will be able to identify.:

- LO1. able to design Finite Automata machines for given problems;
 - LO2. able to analyze a given Finite Automata machine and find out its Language;
 - LO3. able to create Pushdown Automata machine for given CF language(s);
 - LO4. able to generate the strings/sentences of given context-free languages using its grammar;
 - LO5. Able to design Turing machines for given Apply to identify Interpretational problem.
-

Unit I

Languages: Alphabets, string, language, Basic Operations on language, Concatenation, Kleene Star. Finite Automata and Regular Languages: Regular Expressions, Transition Graphs, Deterministic and non-deterministic finite automata, NFA to DFA Conversion, Regular languages and their relationship with finite automata, Pumping lemma and closure properties of regular languages.

Unit II

Context-free languages: Context-free grammars, parse trees, ambiguities in grammars and languages, Pushdown automata (Deterministic and Non-deterministic), Pumping Lemma, Properties of context-free languages, normal forms.

Unit III

Turing Machines and Models of Computations: RAM, Turing Machine as a model of computation, Universal Turing Machine, Language acceptability, decidability, halting problem, Recursively enumerable and recursive languages, unsolvability problems.

Recommended Readings

1. Daniel I.A., Cohen, Introduction to computer theory, John Wiley.
2. Lewis & Papadimitriou, Elements of the theory of computation, PHI.

Suggested Readings

Masters in Computer Science (Semester System)

1. Hopcroft, Aho, Ullman, Introduction to Automata Theory, Language & Computation, Pearson Education.
2. P. Linz, An Introduction to Formal Language and Automata, 4th edition Jones Bartlett Publication .

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MS-CO-EO-306(a)

Paper Name: Data Analysis Using R

Course Objectives:

- CO1. To use Jupyter Notebook for interactive computation
- CO2. To practice Python features such as lists, dictionaries, and files for the given problem
- CO3. To use NumPy functions for array processing
- CO4. To apply Pandas Dataframe for data wrangling
- CO5. To generate graphs for the given data using Matplotlib
- CO6. To understand the basics of R programming in terms of constructs, control statements, string Functions.
- CO7. To understand the use of R for Data analytics.
- CO8. To conduct your independent data analysis.
- CO9. To be able to appreciate and to apply the R programming from a statistical perspective.

Learning Outcomes:

After completing this course, students will be able to:

- LO1. Use Jupyter Notebook for interactive computation
 - LO2. Practice Python features such as lists, dictionaries, and files for the given problem
 - LO3. Use NumPy functions for array processing
 - LO4. Apply Pandas Dataframe for data wrangling
 - LO5. Generate graphs for the given data using Matplotlib
 - LO6. Understand the basics of R programming in terms of constructs, control statements, string Functions.
 - LO7. Understand the use of R for Data analytics.
 - LO8. Conduct your independent data analysis.
 - LO9. Able to appreciate and apply the R programming from a statistical perspective.
-

Unit I

Foundations for data analysis-matrices, the notion of probability, the concept of random variables and various distributions, mean, variance, covariance, normal distributions, an overview of sampling, hypothesis testing, confidence interval, the concept of optimization.

Unit II

installation of R, data editing, use of R as a calculator; functions, and assignments. matrix operations, logical operators, Conditional executions and loops, data management with sequences, repeats, sorting and ordering, lists, vector indexing, factors; display and formatting of strings.

Unit III

Working with data frames, Importing data files; Graphics and plots; basic statistical functions for central tendency, variation, box plots, skewness and kurtosis, correlations; overview of using R functions for simple hypothesis testing, Applications of R.

Recommended Readings

1. Garrett Grolemond, Hands-On Programming with R, O'Reilly Publishers.
2. R for Beginner - https://cran.r-project.org/doc/contrib/Paradis-rdebuts_en.pdf

Suggested Readings

3. A Learning Guide to R - https://www.westernsydney.edu.au/_data/assets/pdf_file/0011/830909/Rnotes_20180905_w eb.pdf

Masters in Computer Science (Semester System)

4. Douglas Montgomery, Applied Statistics And Probability For Engineers, John Wiley & Sons Inc.
5. C.R. Kothari, Research Methodology: Methods And Techniques, New Age International Publishers.
6. Montgomery, Douglas C, Design and Analysis of Experiments, Wiley India.

Masters in Computer Science (Semester System)

F

Paper Code:FS-COMP-MS-C-EO-306(b)

Paper Name : LaTeX: a document preparation system

Course Objectives:

- CO1. To apply various Excel tools and add-ins for analyzing Business problems.
- CO2. To compare mathematical formulas with Spreadsheet formulas
- CO3. To explore, query, and summarize business data.
- CO4. To apply descriptive statistical measures for business decisions.
- CO5. To perform progression analysis and forecasting techniques.
- CO6. To understand how to write documents containing mathematical formulas.
- CO7. To understand how to write articles in different journal styles.
- CO8. To understand how to create PPT in a more presentable manner.
- CO9. To understand how to create using built-in templates.

Learning Outcomes:

After completing this course, students will be able to:

- LO1. Apply various Excel tools and add-ins for analyzing Business problems.
 - LO2. Compare mathematical formulas with Spreadsheet formulas
 - LO3. Explore, query, and summarize business data.
 - LO4. Apply descriptive statistical measures for business decisions.
 - LO5. Perform progression analysis and forecasting techniques.
 - LO6. Understand how to write documents containing mathematical formulas.
 - LO7. Understand how to write articles in different journal styles.
 - LO8. Understand how to create PPT in a more presentable manner.
 - LO9. Understand how to create using built-in templates.
-

Unit I

Installation of the software LaTeX, Structure of LaTeX documents; Special Characters, Producing equations, Matrices, Tables, itemized lists, hypertext links; Page Layout –Title, Abstract, Chapters, Sections, References.

Unit II

Including graphics, images, floating bodies; Producing basic mathematical graphics like line segments, arrows, circles, ovals, Generating index and bibliography, creating a PDF file.

Unit III

Adding a new command; generating spaces, colored text; Writing a sample resume, question paper, article/ research paper; Creating a presentation using beamer.
of part A, B and C are 50, 200 and 500 respectively

Recommended Readings

1. Leslie Lamport, LaTeX: A Document Preparation System, Addison- Wesley.

Suggested Readings

2. Stefan Kottwitz, LaTeX Beginner's Guide, Packt Publishing Limited.
3. Tobias Oetiker, Hubert Partl, Irene Hyna, and Elisabeth Schegle, The Not So Short Introduction to LaTeX 2e, <https://tobi.oetiker.ch/lshort/lshort-a5book.pdf>, 2014.

Masters in Computer Science (Semester System)

Masters in Computer Science (Semester System)

Semester IV

Paper Code:FS-COMP-MS-C-CC-401

Paper Name: Computer Graphics & Multimedia

Course Objectives:

- CO1. To develop line and circle generation algorithms
- CO2. To apply 2D and 3D transformations
- CO3. To develop clipping algorithms for point, line, and polygons
- CO4. To learn the concepts of projections, viewing, and graphics pipeline
- CO5. To create a simple animation and interaction for multimedia presentation
- CO6. To understand image types and color models
- CO7. To describe the concepts regarding the digitization of audio signals
- CO8. To compress images, videos, and audios using data compression methods
- CO9. To encode videos and audios using MPEG
- CO10. To ExplainfunctionalIdentify the core concepts of computer graphics, including viewing, projection, perspective, modeling, and transformation in two and three dimensions.
- CO11. To apply the concepts of color models, lighting and shading models, textures, ray tracing, hidden surface elimination, anti-aliasing, and rendering.
- CO12. To interpret the mathematical foundation of the concepts of computer graphics.
- CO13. To describe the fundamentals of animation, parametric curves, and surfaces, and spotlighting.
- CO14. To identify a typical graphics pipeline and apply graphics programming techniques to design and create computer graphics.
- CO15. To create effective OpenGL programs to solve graphics programming issues, including 3D transformation, object modeling, color modeling, lighting, textures, and ray tracing.
- CO16. To understand multimedia concerning any applications, including business, schools, home, education, and virtual reality.
- CO17. To understand the hardware and software needed to create projects using creativity and organization to create them.
- CO18. To develop multimedia skills to be the principal player of individual multimedia teams in developing projects.
- CO19. To work with all aspects of images.
- CO20. To work with all aspects of sound.
- CO21. To work with all aspects of the video.
- CO22. To learn copyright laws associated with multimedia.
- CO23. To learn the cost involved in multimedia planning, designing, and producing.
- CO24. To learn ways to present their multimedia projects.

Learning Outcomes:

After completing this course, students will be able to:

- LO1. Develop line and circle generation algorithms
- LO2. Apply 2D and 3D transformations
- LO3. Develop clipping algorithms for point, line, and polygons
- LO4. Learn the concepts of projections, viewing, and graphics pipeline
- LO5. Create a simple animation and interaction for multimedia presentation
- LO6. Understand image types and color models
- LO7. Describe the concepts regarding the digitization of audio signals
- LO8. Compress images, videos, and audios using data compression methods
- LO9. Encode videos and audios using MPEG
- LO10. ExplainfunctionalIdentify the core concepts of computer graphics, including viewing, projection, perspective, modeling, and transformation in two and three dimensions.
- LO11. apply the concepts of color models, lighting and shading models, textures, ray tracing, hidden surface elimination, anti-aliasing, and rendering.

Masters in Computer Science (Semester System)

- LO12. interpret the mathematical foundation of the concepts of computer graphics.
- LO13. Describe the fundamentals of animation, parametric curves, and surfaces, and spotlighting.
- LO14. Identify a typical graphics pipeline and apply graphics programming techniques to design and create computer graphics.
- LO15. Create effective OpenGL programs to solve graphics programming issues, including 3D transformation, object modeling, color modeling, lighting, textures, and ray tracing.
- LO16. Students will understand multimedia concerning any applications, including business, schools, home, education, and virtual reality.
- LO17. Students will understand the hardware and software needed to create projects using creativity and organization to create them.
- LO18. The student will develop multimedia skills to be the principal player of individual multimedia teams in developing projects.
- LO19. Students will work with all aspects of images.
- LO20. Students will work with all aspects of sound.
- LO21. Students will work with all aspects of the video.
- LO22. Students will learn copyright laws associated with multimedia.
- LO23. Students will learn the cost involved in multimedia planning, designing, and producing.
- LO24. Students will learn ways to present their multimedia projects.

Unit I

Basic elements of Computer Graphics, Graphics display devices, Applications of Computer Graphics, Raster and random scan; Color Models: RGB, CMY, HSV; Graphics Standard: OpenGL; Scan Conversion: DDA line algorithm, Midpoint circle Algorithm. 2D Transformation: Translation, Rotation, Scaling, Homogeneous Coordinates and Matrix Representation of 2D Transformation, Composite Transformation.

Unit II

3D Graphics: Matrix Representation of 3D transformations, Translation, Rotation, Scaling, Composite Transformation. Overview of concepts: Clipping, orthographic and parallel projection, hidden surface removal, lighting, transparency, modeling and texturing, rendering; Animations: Principles of animations, keyframing, the concept of 2D and 3D animation.

Unit III

Blender: GUI Interface, Selecting, rotating, and Translating Objects, Using Snap to move objects precisely, Creating mesh primitives and extrusions, Subdividing meshes, Creating a simple creature, Joining mesh objects and stitching vertices, Organizing a scene with layers, groups, and hierarchies, Assigning glossy and reflective materials to objects, Creating bump maps, Creating sky and ambient light, Understanding ambient occlusion, Adding motion blur and depth of field, Editing animation in the Graph Editor, Building and animating a simple character.

Recommended Readings

1. Foley, van Dam, Feiner and Hughes, Computer Graphics (Principles and Practice), Addison Wesley (Indian Edition).
2. D Hearn and PM Baker, Computer Graphics, Prentice Hall of India (Indian Edition).

Suggested Readings

3. DF Roger, Mathematical Elements for Computer Graphics.
4. Krishnamurthy N, Introduction to Computer Graphics, Tata McGraw Hill.
5. Zhigang X. and Plastock Ra, Theory and Problems of Computer Graphics (Schaum's Outline), Tata McGraw Hill.

Web Resources

1. <https://www.cs.duke.edu/brd/Teaching/Previous/Animation/animation.html>

Masters in Computer Science (Semester System)

2. [http://zikky.lecturer.pens.ac.id/Produksi 3D untuk Designer/Beginning Blender-book.pdf](http://zikky.lecturer.pens.ac.id/Produksi%203D%20untuk%20Designer/Beginning%20Blender-book.pdf)
3. <http://www.blenderhd.com/wp-content/uploads/2015/08/BeginnersGuideToBlender.pdf>
4. https://people.sc.fsu.edu/~gerlebacher/gd/blender/blender/blender_noob_to_pro.pdf
5. [http://download.blender.org/documentation/pdf/John M Blain - An Introduction To Blender 3D - A Book For Beginners \(2011\).pdf](http://download.blender.org/documentation/pdf/John%20M%20Blain%20-%20An%20Introduction%20To%20Blender%203D%20-%20A%20Book%20For%20Beginners%20(2011).pdf)
6. http://www.cdschools.org/cms/lib04/PA09000075/Centricity/Domain/81/BlenderBasics_4thEdition2011.pdf
7. <https://docs.blender.org/manual/en/dev/index.html>

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MS-C-CC-402

Paper Name: Android Programming

Course Objectives:

- CO1. To create an android project from XML Layout.
- CO2. To debug Android apps and create UI fragments
- CO3. To pass data between fragments
- CO4. To design apps with audio playback.
- CO5. To create a database and communicate with mobile apps
- CO6. To install and configure Android application development tools
- CO7. To design and develop user interfaces for the Android platform.
- CO8. To save state information across important operating system events.
- CO9. To apply Java programming concepts to Android application development.
- CO10. To develop the ability to develop Android Application

Learning Outcomes:

After completing this course, students will be able to:

- LO1. Create an android project from XML Layout.
 - LO2. Debug Android apps and create UI fragments
 - LO3. Pass data between fragments
 - LO4. Design apps with audio playback.
 - LO5. Create database and communicate with mobile apps
 - LO6. Install and configure Android application development tools.
 - LO7. Design and develop user interfaces for the Android platform.
 - LO8. Save state information across important operating system events.
 - LO9. Apply Java programming concepts to Android application development.
 - LO10. Develop the ability to develop Android Application
-

Unit -I

Introduction: What is Android?, Android Architecture, Setting Android Environment, Android SDK Manager & required Packages, Using Android Studio, Android Virtual Device(AVD), Creating First Android Application, Package Structure, Introduction to Gradle, Running the Application, Views, Layouts and more.

Unit – II

Introduction to Views: TextView, EditText View, RadioButton and CheckBox View, Button View, ImageView and ImageButton View, Toast, Notifications.

Introduction to Layouts/ViewGroups: Linear Layout, Relative Layout, Tabular Layout, Hierarchical Layout Arrangements, Adapter, and Adapter View, Using ListView and GridView, SQLite Database.

Unit – III

Spinner in Android, Working with Spinners, Margin and Padding, Working with EditText and TextView, RadioGroup, RadioButton and CheckBox, AutoCompleteTextView in Android, Android Core, and Projects.

Location-Based Services: Sending Email, Sending SMS, Phone Calls

Activity in Android, Intents in Android, Introduction to Fragments, Working with Fragments

Masters in Computer Science (Semester System)

Recommended Readings

1. Android Programming for Beginners by John Horton Publisher: Packt Publishing
2. Learn Java for Android Development (2nd edition) by Jeff Friesen Publisher: Apress

Suggested Readings

3. James C. Sheusi, Android application development for java programmers, Cengage Learning.
4. Jerome F. DiMarzio, Beginning Android Programming with Android Studio, Fourth Edition, John Wiley & Sons.
5. Kristin Marsicano , Chris Stewart , Bill Phillips, Programming: The Big Nerd Ranch Guide, Big Nerd Ranch Guides.

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MS-C-CE-403(a)

Paper Name : Cloud Computing

Objective – After completing this course the student will have an understanding of key aspects of cloud computing

Unit I

Introduction to Cloud Computing, Services provided by cloud-SaaS, PaaS, IaaS, DaaS, etc. Functioning of cloud computing, Advantages, Disadvantages, Applications, Cloud Service Providers- Amazon AWS, Google App Engine, Microsoft, VMware. Virtualization concepts, Objectives, Types of Virtualization & its benefits, Introduction to Various Virtualization OS (Hypervisor). Virtualization for Enterprises

Unit II

Designing and Implementing a Data Center-Based Cloud, Industry and International Standards for Cloud Implementation, Building private cloud using open source tools, Integration of Public and Private Cloud. Private, Public & Hybrid Clouds, their Advantages & Disadvantages, On-Premises and Off-Premises Cloud services, installing a Cloud service.

Unit III

Cloud Security issues - Infrastructure Security, Network level security, Host level security, Application-level security, Data privacy and security Issues, Jurisdictional issues raised by Data location, Access Control, Trust, Reputation, Risk, and Authentication in cloud computing

Suggested Readings

1. Thomas Erl, Cloud Computing Concepts Technology and Architecture, Prentice Hall.
2. Rajkumar Buyya, James Broberg and Andrzej Goscinski, Cloud Computing Principles and paradigms, John Wiley and Sons, Inc. Publication.
3. Dan C. Marinescu, Cloud Computing Theory and Practice, Morgan Kaufman Publication.

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MS-C-CE-403(b)

Paper Name: Internet of Things

Course Objectives:

- CO1. To understand the definition and significance of the Internet of Things
- CO2. To discuss the architecture, operation, and business benefits of an IoT solution
- CO3. To examine the potential business opportunities that IoT can uncover
- CO4. To explore the relationship between IoT, cloud computing, and big data
- CO5. To identify how IoT differs from traditional data collection systems
- CO6. To understand the definition and significance of the Internet of Things
- CO7. To discuss the architecture, operation, and business benefits of an IoT solution
- CO8. To examine the potential business opportunities that IoT can uncover
- CO9. To explore the relationship between IoT, cloud computing, and big data
- CO10. To identify how IoT differs from traditional data collection systems.

Learning Outcomes:

After completing this course, students will be able to:

- LO1. Understand the definition and significance of the Internet of Things
 - LO2. Discuss the architecture, operation, and business benefits of an IoT solution
 - LO3. Examine the potential business opportunities that IoT can uncover
 - LO4. Explore the relationship between IoT, cloud computing, and big data
 - LO5. Identify how IoT differs from traditional data collection systems
 - LO6. Understand the definition and significance of the Internet of Things
 - LO7. Discuss the architecture, operation, and business benefits of an IoT solution
 - LO8. Examine the potential business opportunities that IoT can uncover
 - LO9. Explore the relationship between IoT, cloud computing, and big data
 - LO10. Identify how IoT differs from traditional data collection systems.
-

Unit I

M2M to IoT: Introduction, Market Perspective, Architectural Overview. M2M to IoT Technology- Devices and gateways, Local and wide area networking, Data management, Business processes in IoT, IoT analytics, Knowledge management, IOT Architecture, Architecture Reference Model, Real-world design constraints.

Unit II

IoT Use Cases- Asset Management, **Industrial Automation**- Service-oriented architecture-based device integration, SOCRADES: realizing the enterprise integrated Web of Things, IMC-AESOP: from the Web of Things to the Cloud of Things, **Commercial Building Automation**- Introduction, Case study: phase one-commercial building automation today, Case study: phase two- commercial building automation in the future.

Unit III

Internet of Things Privacy, Security and Governance Introduction, Overview of Governance, Privacy and Security Issues, Contribution from FP7 Projects, Security, IOT and Smart Cities, Privacy and Trust in IoT-Data-Platforms for Smart Cities, First Steps Towards a Secure Platform, Smartie Approach. Data Aggregation for the IoT in Smart Cities, Security

Recommended Readings

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.

Masters in Computer Science (Semester System)

2. Vijay Madiseti and Arshdeep Bahga, Internet of Things (A Hands-on-Approach), 1stEdition, VPT.
3. Francis daCosta, Rethinking the Internet of Things: A Scalable Approach to Connecting Everything, 1st Edition, Apress Publications.
4. Hakim Cassimally, Designing the Internet of Things , Adrian McEwen (Author).
5. Dr. Ovidiu Vermesan, Dr. Peter Friess. Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems, River Publishers.
6. Vijay Madiseti , Arshdeep Bahga, Internet of Things, A Hands-on-Approach.
7. Daniel Minoli, Building the internet of things with ipv6 and mipv6, The Evolving World of M2M Communications, John Wiley & Sons.

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MS-CE-403(c)

Paper Name: Big Data & Data Mining

Course Objectives:

- CO1. To explain characteristics and use cases and applications of Big Data
- CO2. To develop MapReduce operation using Hadoop
- CO3. To be able to understand the role of Virtualization Technologies
- CO4. To design and implement systems for data mining.
- CO5. To evaluate the performance of different data-mining algorithms.
- CO6. To propose data-mining solutions for various applications.

Learning Outcomes:

After completing this course, students will be able to:

- LO1. Explain characteristics and use cases and applications of Big Data
 - LO2. Develop MapReduce operation using Hadoop
 - LO3. Ability to understand the role of Virtualization Technologies
 - LO4. design and implement systems for data mining.
 - LO5. Evaluate the performance of different data-mining algorithms.
 - LO6. Propose data-mining solutions for various applications.
-

Unit I

Data mining Introduction: Definition, Data mining tasks, Data mining as a step of the Knowledge discovery process, Applications of Data mining; Data objects and types of attributes, Recalling mean, median, mode, and weighted arithmetic mean, Data quality, an overview of data preprocessing.

Unit II

Classification analysis- definition, Overview of various classification techniques; Decision tree induction- working, examples, specifying attribute test conditions, Measures of node impurity, measures for selecting best split; Evaluating the performance of a classifier- Holdout method, Random subsampling, cross-validation, Bootstrap.

Unit III

Association analysis: support, confidence, association rules, Frequent Itemsets; Frequent itemset generation - Apriori principle, Apriori algorithm, and examples, FP growth algorithm, and examples; Closed and maximal frequent itemsets. Cluster analysis: Definition, an overview of basic clustering methods, Density-based methods-DBSCAN.

Recommended Readings

1. Jiawei Han and Micheline Kamber, Data Mining: Concepts and Techniques, 3rd edition.
2. Pang-Ning Tan, Michael Steinbach, Vipin Kumar, Introduction to Data Mining, Pearson Education.

Suggested Readings

3. Richard Roiger, Michael Geatz, Data Mining: A Tutorial Based Primer, Pearson Education.
4. G.K. Gupta, Introduction to Data Mining with Case Studies, PHI.
5. Soman K. P., Diwakar Shyam, Ajay V., Insight into Data mining: Theory and Practice, PHI.

Masters in Computer Science (Semester System)

6. Witten, Frank, Data Mining:: Practical Machine Learning Tools and Techniques (Morgan Kaufmann Series in Data Management Systems) Prentice Hall.

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MSC-CS-CE-403(d)

Paper Name: Machine Learning

Course Objectives:

- CO1. To be able to design Finite Automata machines for given problems;
- CO2. To be able to analyze a given Finite Automata machine and find out its Language;
- CO3. To be able to create Pushdown Automata machine for given CF language(s);
- CO4. To be able to generate the strings/sentences of given context-free languages using its grammar;
- CO5. To be able to design Turing machines for given Apply to identify Interpretational problem.

Learning Outcomes:

After completing this course, students will be able to-

- LO1. Able to design Finite Automata machines for given problems;
 - LO2. Able to analyze a given Finite Automata machine and find out its Language;
 - LO3. Able to create Pushdown Automata machine for given CF language(s);
 - LO4. Able to generate the strings/sentences of given context-free languages using its grammar;
 - LO5. Able to design Turing machines for given Apply to identify Interpretational problem.
-

Unit I

Introduction: Concept of Machine Learning, Applications of Machine Learning, Key elements of Machine Learning, Supervised vs. Unsupervised Learning, Statistical Learning: Bayesian Method, The Naive Bayes Classifier. Tools for Machine Learning and Linear Algebra Overview: Plotting of Data, Vectorization, Matrices, and Vectors: Addition, Multiplication, Transpose and Inverse using available tools/libraries with Python.

Unit II

Linear Regression: Prediction using Linear Regression, Gradient Descent, Linear Regression with one variable, Linear Regression with multiple variables, Polynomial Regression, Feature Scaling/Selection. Logistic Regression: Classification using Logistic Regression, Logistic Regression vs. Linear Regression, Logistic Regression with one variable and with multiple variables.

Unit III

Regularization: Regularization and its utility: The problem of Overfitting, Application of Regularization in Linear and Logistic Regression, Regularization and Bias/Variance. Neural Networks: Introduction, Model Representation, Gradient Descent vs. Perceptron Training, Stochastic Gradient Descent, Multilayer Perceptrons, Multiclass Representation, Backpropagation Algorithm.

Recommended Readings

1. Tom M. Mitchell, Machine Learning, First Edition, Tata McGraw-Hill Education.
2. Ethem Alpaydin, Introduction to Machine Learning, 2nd Edition, The MIT Press.

Masters in Computer Science (Semester System)

3. Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer.
4. Mevin P. Murphy, Machine Learning: A Probabilistic Perspective, The MIT Press.

Suggested Readings

1. John Paul Mueller, Luca Massaron, Machine Learning For Dummies, For Dummies; 1st edition.
2. O Theobald, Machine Learning for Absolute Beginners: A Plain English Introduction, Scatterplot Press; 2nd edition.
3. Andreas C. Müller, Sarah Guido, Introduction to Machine Learning with Python: A Guide for Data Scientists, O'Reilly; 1st edition
4. <https://www.cmpe.boun.edu.tr/~ethem/i2ml3e/>

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MS-C-EO-406(a)

Paper Name: Natural Language Processing

Course Objectives:

- CO1. To have an introduction of the fundamental concepts and techniques of natural language processing (NLP).
- CO2. To gain an in-depth understanding of the computational properties of natural languages and the commonly used algorithms for processing linguistic information.
- CO3. To examine NLP models and algorithms using both the traditional symbolic and the more recent statistical approaches.
- CO4. To understand critical concepts from NLP are used to describe and analyze language.
- CO5. To perform POS tagging and context-free grammar for the English language.
- CO6. To understanding semantics and pragmatics of English language for processing.
- CO7. To write programs in Python to carry out natural language processing

Learning Outcomes:

After completing this course, students will be able to-

- LO1. Introduction to the fundamental concepts and techniques of natural language processing (NLP).
 - LO2. Students will gain an in-depth understanding of the computational properties of natural languages and the commonly used algorithms for processing linguistic information.
 - LO3. The course examines NLP models and algorithms using both the traditional symbolic and the more recent statistical approaches.
 - LO4. Critical concepts from NLP are used to describe and analyze language.
 - LO5. POS tagging and context-free grammar for the English language.
 - LO6. Understanding semantics and pragmatics of English language for processing.
 - LO7. Writing programs in Python to carry out natural language processing
-

Unit I

Introduction, Basics of text processing, Spelling Correction: Edit Distance; N-Gram Language Models, Evaluation of Language Models, Basic Smoothing, Computational Morphology, Introduction to POS Tagging, Overview of Hidden Markov Model, Basics of Models for Sequential tagging – Introduction to Maximum entropy and Conditional Random Fields.

Unit II

Constituency syntax parsing, examples of parsing using CKY and PCFG, Introduction to Dependency Grammars and Parsing, understanding of Transition Based Parsing; Distributional Semantics - Introduction, Applications; Word Embedding: Frequency-based embedding, Prediction based embeddings. Lexical Semantics: an overview of WordNet, Word Sense Disambiguation.

Unit III

Topic models: introduction, LDA; Introduction to Entity Linking and Information Extraction; Text Summarization: an overview of various approaches; Text Classification: introduction

Masters in Computer Science (Semester System)

and simple practical implementation using Python. Sentiment Analysis: Concept, Analysis, and Applications.

Required Readings

1. James Allen, Natural Language Understanding, Pearson Education; 2nd edition.
2. Jurafsky / Martin. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition, 2e.
3. Nitin Indurkha, Fred J. Damerau, Handbook of Natural Language Processing, Taylor and Francis; Second edition.
4. Alexander Clark, Chris Fox, Shalom Lappin, The Handbook of Computational Linguistics and Natural Language Processing, Wiley-Blackwell; 1st edition
5. Steven Bird, Ewan Klein, Edward Loper, Natural Language Processing with Python: Analysing Text with the Natural Language Toolkit, Shroff pub.
6. Christopher D. Manning, Hinrich Schütze, Foundations of Statistical Natural Language Processing, MIT press.

Suggested Readings

1. Frederick Jelin, Statistical Methods for Speech Recognition (Language, Speech, and Communication) Fourth Printing Edition.
2. Yoav Goldberg, Graeme Hirst, Morgan and Claypool, Neural Network Methods for Natural Language Processing Synthesis Lectures on Human Language Technologies, Life Sciences.

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MS-C-EO-406(b)

Paper Name: Introduction to Cyber Security

Course Objectives:

CO1. To identify and classify various attacks

CO2. To encrypt and decrypt messages using block chippers and signs.

CO3. To create a digital signature using multiple algorithms.

CO4. To describe web security, intruders, viruses, and firewalls

Learning Outcomes:

After completing this course, students will be able to-

LO1. Identify and classify various attacks

LO2. Encrypt and decrypt messages using block chippers and signs.

LO3. Create a digital signature using multiple algorithms.

LO4. Describe web security, intruders, viruses, and firewalls

Unit I

Basics: Linux/Mac Terminal and Commands, Basic Computer Terminology, Computer Security models, Computer Security Terms, Computer Ethics, Business, and Professional Ethics, Need for cyber security; Cyber Frauds and crimes, Digital Payments, Various Search Engines, Introduction to Auditing, Deep Web, VAPT, Smartphone Operating systems, introduction to compliances, Globalization and borderless world.

Unit II

Basic Python Scripting: Python Basics, Variables, and Types, Lists, Basic Operators, String Formatting, Basic String Operations, Conditions, Loops, Functions, Classes and Objects, Dictionaries, Modules, and Packages.

Unit III

Cyber Laws: Need for Cyber Regulations; Scope and Significance of Cyber laws: Information Technology Act 2000; Network and Network Security, Access and Unauthorised Access, Data Security, E Contracts and E Forms. Penal Provisions for Phishing, Spam, Virus, Worms, Malware, Hacking, Trespass, and Stalking; Human rights in cyberspace, International Co-operation in investigating cybercrimes.

Recommended Readings

1. Behrouz A. Forouzan (2004). Data communication and Networking. Tata McGraw-Hill.
2. Kurose, James F. & Ross, Keith W. (2003). Computer Networking: A Top-Down Approach Featuring the Internet (3rd Ed.). Pearson Education.
3. Langtangen, H.P. (2012). Python Scripting for Computational Science (4th Ed.). Springer

Masters in Computer Science (Semester System)

4. Craig, B. (2012). Cyber Law: The Law of the Internet and Information Technology. Pearson. Sharma J. P. & Kanojia S. (2016). Cyber Laws. New Delhi: Ane Books Pvt Ltd.
5. Paintal, D. Law of Information Technology. New Delhi: Taxmann Publications Pvt. Ltd

Suggested Readings

1. Shema, M. (2012). Hacking Web Apps: Detecting and Preventing Web Application Security Problems.
2. <https://uou.ac.in/sites/default/files/slm/Introduction-cyber-security.pdf>
3. Computer Programming And Cyber Security for Beginners: This Book Includes: Python Machine Learning, SQL, Linux, Hacking with Kali Linux, Ethical Hacking. Coding and Cybersecurity Fundamentals, Zach Codings, Independently published

Masters in Computer Science (Semester System)

Paper Code:FS-COMP-MS-C-CC-405

Paper Name: Combined Practical & Project/Dissertation/Industrial Training

Course Objectives:

- CO1. Identify and define the problem statement
- CO2. Define and justify the scope of the proposed problem
- CO3. Gather and analyze system requirements
- CO4. Propose an optimized solution among the existing solutions
- CO5. Practice software analysis and design techniques
- CO6. Develop technical report writing and oral presentation skills
- CO7. Develop a functional application based on the software design
- CO8. Apply to code, debugging, and testing tools to enhance the quality of the software
- CO9. Prepare the proper documentation of software projects following the standard guidelines
- CO10. Become a master in specialized technology
- CO11. Become updated with all the latest changes in the technological world.
- CO12. Ability to communicate efficiently.
- CO13. Ability to be a multi-skilled engineer with sound technical knowledge, management, leadership, and entrepreneurship skills.
- CO14. Capability and enthusiasm for self-improvement through continuous professional development and life-long learning
- CO15. Awareness of the social, cultural, global, and environmental responsibility of an engineer.

Learning Outcomes

After completing this course, students will be able to:

- LO1. Identify and define the problem statement
 - LO2. Define and justify the scope of the proposed problem
 - LO3. Gather and analyze system requirements
 - LO4. Propose an optimized solution among the existing solutions
 - LO5. Practice software analysis and design techniques
 - LO6. Develop technical report writing and oral presentation skills
 - LO7. Develop a functional application based on the software design
 - LO8. Apply to code, debugging, and testing tools to enhance the quality of the software
 - LO9. Prepare the proper documentation of software projects following the standard guidelines
 - L10. Become a master in specialized technology
 - LO11. Become updated with all the latest changes in the technological world.
 - LO12. Ability to communicate efficiently.
 - LO13. Ability to be a multi-skilled engineer with sound technical knowledge, management, leadership, and entrepreneurship skills.
 - LO14. Capability and enthusiasm for self-improvement through continuous professional development and life-long learning
 - LO15. Awareness of the social, cultural, global, and environmental responsibility of an engineer.
-

Practical Training and Project Work:

1. Project Work may be done individually or in groups in case of bigger projects. However, if the project is done in groups, each student must be given responsibility for a distinct module and care should be taken to monitor the individual student.

Masters in Computer Science (Semester System)

2. Project Work can be carried out in the college or outside with prior permission of the college.
3. The Student must submit a synopsis of the project report to the college for approval. The Project Guide can accept the project or suggest modification for resubmission. Only on acceptance of the draft project report, the student should make the final copies.

Submission Copy:

The Student should submit a spiral-bound copy of the project report.

Format of the Project:

1. **Paper:**
The Report shall be typed on White Paper of A4 size.
2. **Final Submission:**
The Report to be submitted must be original.
3. **Typing:**
Font:- Times New Roman
Heading:- 16 pt., Bold
Subheading:- 14 pt, Bold
Content:- 12 pt.
Line Spacing:- 1.5 lines.
Typing Side :-One Side
Font Color:- Black.
4. **Margins:**
The typing must be done in the following margin:
Left : 0.75”
Right: 0.75”
Top: 1”
Bottom: 1”
Left Gutter: 0.5”
5. **Binding:**
The report shall be Spiral Bound.
6. **Title Cover:**
The Title cover should contain the following details:
Top: Project Title in block capitals of 16pt.
Centre: Name of project developer’s and Guide name.
Bottom: Name of the university, Year of submission all in block capitals of 14pt letters on separate lines with proper spacing and centering.
7. **Blank sheets:**
At the beginning and end of the report, two white blank papers should be provided, one for the Purpose of Binding and other to be left blank.
8. **Content:**
 - I). Acknowledgment
 - II). Institute/College/Organization certificate where the project is being developed.
 - III). Table of contents
 - IV). A brief overview of the project
 - V). Profiles of problems assigned
 - VI). Study of Existing System
 - VII). System Requirement
 - VIII). Project plan
 - o Team Structure
 - o Development Schedule

Masters in Computer Science (Semester System)

- Programming language and Development Tools
- IX). Requirement Specification
- X). Design
- Detailed DFD and Structure Diagram
- Data structure, Database and File Specification

- XI). Project Legacy
 - Current Status of project
 - Remaining Areas of concern
 - Technical and Managerial Lessons Learnt
 - Future Recommendations
- Nomenclature and Abbreviations.
- Bibliography
- Source Code

Masters in Computer Science (Semester System)

Teaching-Learning Process

The teaching learning process may include the following-

- Lectures
- Discussions
- Simulations
- Virtual Labs
- Role Playing
- Participative Learning
- Interactive Sessions
- Seminars
- Research-based Learning/ Dissertation/ Case Study/ Project Work

The Blended Learning mode of teaching and learning is preferable in which offline (face-to-face) and online learning both are used to provide learners the opportunity to enjoy both of the worlds. Teachers can share instructions, lecture notes, and assignments online. On the other hand, students can share information/work/assignments with teachers and other students directly in a collaborative setting. This may have a more enriched learning experience, and collaboration between students can be improved upon if group activities rely on information gathered from online resources or lessons. Students who complete online coursework followed by interactive, face-to-face class activities have richer educational experiences.

Assessment and Evaluation

- A comprehensive and continuous evaluation by mid-semester examinations at regular intervals to find out each course level learning outcome
- Formative assessment on the basis of activities of a learner throughout the program instead of one assessment. for this provision of internal exams, student seminars, and assignments is included
- Open book exam is suggested for internal/ mid-term exams to better facilitate the understanding of the knowledge required
- Group examinations are recommended on problem-solving exercises and in major projects to enhance the teamwork capabilities of the learner
- Collaborative/Individual assignments are useful to enhance the capability of learners to gain domain-specific knowledge
- Student Seminars and Quizzes are recommended for the continuous learning and evaluation process

ELIGIBILITY FOR ADMISSION

Graduates possessing **55% marks (As per Admission Policy Govt. of Rajasthan)** in any faculty of any statutory university shall be eligible for admission to the M.Sc. Computer Science Course (Relaxation to SC/ST etc. as per Prevailing Rules)

PASS CRITERIA

For passing in the examination, a candidate is required to obtain at least a Satisfactory Grade in each paper (Internal + External) and also acquire a Satisfactory Grade in theory and practical separately (in each semester examination).

Masters in Computer Science (Semester System)

INSTRUCTIONS TO PAPER SETTER

The question paper contains 3 sections. **Section-A** consists of 10 questions (at least 3 questions from each unit). **Section-B** will consist of 9 questions (3 questions from each unit). **Section C** will consist of 6 questions (2 questions from each unit). The word limit of parts A, B, and C are 50, 200, and 500 respectively

INSTRUCTIONS FOR PRACTICAL EXAMINATION

Marks Distribution for Practical Exam -

Each practical exam is to be conducted by two examiners one External and one Internal. The external examiner should be a senior lecturer from the jurisdiction of other universities. Credit Weightage distribution for external practical of 4 credits is as under

- a) Practical Examination exercise of 3 questions 2 credits
- b) Viva-Voce 1 credit
- c) Laboratory Exercise File 1 credit

Marks distribution for External Project report of 40 marks is as under

External Evaluation-

- Research Project/ Case Study 2 credits
- Presentation 1 credit
- External Viva Voce 1 credit

Internal Evaluation- Dissertation 1 credit

INSTRUCTIONS FOR STUDENTS

The student has to complete two months of career-oriented summer training from any firm/organization. If the student does not get a chance to go for training, he/she can choose a research topic and can complete the dissertation under the supervision of any of the faculty in his college.

The student who has to opt for training has to provide a signed certificate from the firm/organization authority stating that the student has spent two months as a trainee in his organization/firm. The student who has opted for a dissertation has to submit his/her dissertation report with a certificate from his supervisor.

In both cases, the student has to present his work in front of all the faculty members and fellow students at the starting of the next session.

In terms of credits, every one-hour session of L amounts to 1 credit per semester and a minimum of two-hour sessions of T or P amounts to 1 credit per semester.

*** An Academic/ Industrial Tour shall be organized by the college/department in every session. A Tour Report shall be prepared and submitted by the students after a study tour to industries/academic institutions of repute.**

EVALUATION

Internal Assessment -	Midterm Examination	10%
	Term paper	10%
	Students Participation	5%
External Assessment-	75%	

Examination Paper Pattern

The question paper contains 3 sections. **Section-A** consists of 10 questions (at least 3 questions from each unit of syllabus). **Section-B** will consist of 9 questions (3 questions from each unit). **Section-C** will consist of 6 questions (2 questions from each unit). The word limit of part A, B and C are 50, 200 and 500 respectively

Masters in Computer Science (Semester System)

Key Features of Revised Curriculum

Following are the key features of the revised curriculum-

- Student Centric Teaching and Learning approach
- Technology oriented approach of teaching
- Hand-on Practical/ Laboratory Sessions
- Problem-oriented teaching and learning
- Problem-analysis oriented assignments and evaluation
- Enhance logical thinking and analytical capabilities

Appendices

List of Open Electives offered by the University -