GANGA INSTITUTE OF TECHNOLOGY AND MANAGEMENT, KABLANA (JHAJJAR)

An Autonomous Institute (UGC)

Approved by AICTE, New Delhi and Affiliated to MDU, Rohtak

NAAC 'A' GRADE

Scheme of Studies and Examination
Master of Computer Science and Applications

1. DEFINITION OF CREDIT

1	1 Lecture (L) per week	1 Credit
2	1 Tutorial (T) per week	1 Credit
3	1 Practical (P) per week	0.5 Credit
4	2 Practical (Lab) per week	1 Credit

2. RANGE OF CREDIT

A range of credits from 110-120 for a student will be eligible to get Post Graduate degree.

3. STRUCTURE OF MASTER OF COMPUTER APPLICATIONS PROGRAM (MCA)

Sr. No.	Category	Breakup of Credits MCA
1	Discipline-Specific Courses (DSC) /Experiential Learning	72*
2	Discipline-Specific Elective Courses	32*
3	Multidisciplinary Open Elective Courses	03*
4	Foundation Elective Courses	03*
4	Research Project/Dissertation	06*
	Total Credits	**116***

Minor variation is allowed as per need of the respective disciplines.

4. COURSE CODE AND DEFINITIONS

Sr. No.	Category	Course Code
1	Discipline Specific Courses	DSC
2	Discipline Specific Elective Courses	DSEC
3	Experiential Learning	EL
4	Research Report	RR
5	Multidisciplinary Open Elective Courses	MDC
6	Bridge Course	BC

5. ELIGIBILITY FOR ADMISSION TO MCA 2-YEARPROGRAMME

a) Passed BCA/B.Sc. (Hons.) Computer Science/ B.E. or B.Tech. (CSE/IT)/ B.Voc. (Software Development/IT) or an equivalent degree with having at least 50% marks (45% for SC/ST candidates of Haryana only) in aggregate.

Or

b) Passed B.Sc. / B. Com/ B.A with Mathematics at 10+2 level or at Graduation level with having at least 50% marks (45% for SC/ST candidates of Haryana only)in aggregate, along with the students admitted with this eligibility will have to simultaneously undertake additional

c) Bridge Course* as prescribed by the institute during the first semester.

Note: It is compulsory for each student to pass out Bridge Course (three additional theory papers and one practical as prescribed in scheme of examination of Bridge Course) as per institute norms during the 1st year of MCA-2 year course and the degree will be awarded after the completion of Bridge Course. However, these papers under Bridge Course will be taught only in the 1st semester of the course.

Note:

1. The duration of all the end-term theory examinations shall be 3 hours.
2. The Criteria for Assessment of Theory Courses throughout the Program shall be as under:
   a) Sessional Examination : 60%.
   b) Assignments/Presentations/Seminars and Class Participation : 20%
   c) Attendance : 20%
      Less than 65% : 0% of Attendance marks
      65%-69.99% : 40% of Attendance marks
      70%-74.99% : 60% of Attendance marks
      75%-80% : 80% of Attendance marks
      Above 80% : 100% of Attendance marks
3. The Criteria for awarding the Internal assessment practical course throughout the Program shall be as under:
   a) Practical Assignments/Practical File : 60% Weightage of Assessment
   b) Attendance (Criteria as mentioned above in 2(c)) : 40% Weightage of Assessment
4. The panel of examiners for end-semester theory examinations shall be prepared and approved by the Department of Computer Science and Application of the internal/external examiners based on their expertise/specialization/area of interest.
5. The panel of examiners based on the examiners' expertise/specialization/area of interest for practicum/viva-voce examination shall be recommended by the Department of Computer Science and Application. In case of unavailability of external examiners due to unavoidable circumstances, the Controller of Examinations may allow the conduct of practical examinations by the internal examiners.
6. The panel of examiners for assessment of the Project Reports/Dissertation/Research Project/Summer Internship Reports shall be recommended by the Department of Computer Science and Application.
7. A student while selecting the minor discipline has to ensure that the courses of the opted minor discipline do not match with the major courses of his/her opted programme.
8. The student may choose the Multidisciplinary Courses from other than the courses offered by their own/respective department.

Scheme of Studies and Examination
MCA – 1st Semester
w.e.f. 2024-25

SN	Category	Course Code	Course Title	Hours per week			Total Load Per Week	Credits	Examination Schedule (Marks)				Exam Duration in Hours
				L	T	P			Assessment	End Semester Exam		Total	
										Theory	Practical		
1	Discipline-Specific Courses	DSC-MCA-101A	Object Oriented Programming Using JAVA	4	0	0	4	4	40	60		100	3
2	Discipline-Specific Courses	DSC-MCA-103A	Compiler Design	4	0	0	4	4	40	60		100	3
3	Discipline-Specific Courses	DSC-MCA-105A	Computer Graphics and Multimedia	4	0	0	4	4	40	60		100	3
4	Discipline Specific Elective Courses	Refer Table-I(a)	Elective-I	4	0	0	4	4	40	60		100	3
5	Discipline Specific Elective Courses	Refer Table-I(b)	Elective-II	4	0	0	4	4	40	60		100	3
6	Experiential Learning	EL-MCA-111A	Software Lab-1	0	0	6	6	3	50		50	100	3
7	Experiential Learning	EL-MCA-113A	Software Lab-2	0	0	6	6	3	50		50	100	3
Total								**26**	300	300	100	**700**	

GANGA INSTITUTE OF TECHNOLOGY AND MANAGEMENT, JHAJJAR (HR.), DELHI-NCR

Scheme of Studies and Examination

Bridge Course

w.e.f. 2024-25

SN	Category	Course Code	Course Title	Hours per week			Total Load Per Week	Credits	Examination Schedule (Marks)				Exam Duration in Hours
				L	T	P			Assessment	End Semester Exam		Total	
										Theory	Practical		
1	Bridge Course	BC-MCA-111A	Computer Fundamentals and Programming in C	4	0	0	4	4	40	60		100	3
2	Bridge Course	BC-MCA-121A	C++ and Data Structures	4	0	0	4	4	40	60		100	3
3	Bridge Course	BC-MCA-123A	Visual Basic and Database Systems	4	0	0	4	4	40	60		100	3
4	Bridge Course- LAB	BC-MCA-125A	Software Lab-Bridge Course	0	0	8	8	4	50		50	100	3
							Total	**16**	170	180	50	**400**	

Note: It is compulsory for each student to pass Bridge Courses (three additional theory papers and one practical as prescribed in scheme of examination of Bridge Course) as per Institute norms during the 1st year of MCA-2 year course and the degree will be awarded after the completion of Bridge Courses. However, these papers under Bridge Courses will be taught only in the 1st semester of the course.

Scheme of Studies and Examination
MCA – 2nd Semester
w.e.f. 2024-25

SN	Category	Course Code	Course Title	Hours per week			Total Load Per Week	Credits	Examination Schedule (Marks)				Exam Duration in Hours
				L	T	P			Assessment	End Semester Exam		Total	
										Theory	Practical		
1	Discipline-Specific Courses	DSC-MCA-102A	Advance Object Technology	4	0	0	4	4	40	60		100	3
2	Discipline-Specific Courses	DSC-MCA-104A	Advance Database Systems and Data Warehouse	4	0	0	4	4	40	60		100	3
3	Discipline-Specific Courses	DSC-MCA-106A	Operating Systems and Shell Programming	4	0	0	4	4	40	60		100	3
4	Discipline Specific Elective Courses	Refer Table-I (c)	Elective-III	4	0	0	4	4	40	60		100	3
5	Discipline Specific Elective Courses	Refer Table-I (d)	Elective-IV	4	0	0	4	4	40	60		100	3
6	Experiential Learning	EL-MCA-108A	Software Lab-3	0	0	6	6	3	50		50	100	3
7	Experiential Learning	EL-MCA-110A	Software Lab-4	0	0	6	6	3	50		50	100	3
8	Research Report	RR-MCA-112A	Industry Internship Report/ Project Report/Dissertation–I	0	0	0	6	3	50		50	100	3
9	Foundation Elective Courses	Refer Table-III	Refer Pool of Foundation Elective courses defined by the Institute	3	0	0	3	3	40	60		100	3
							Total	**32**	380	360	160	**900**	

Table No. I (a) (Discipline Specific Elective Courses)
Elective- I

Sr. No.	Category	Course Code	Course Title
1	Discipline Specific Elective Courses	DSEC-MCA-111A	Digital Design & Computer Architecture
2		DSEC-MCA-113A	Cloud Computing

Table No. I (b) (Discipline Specific Elective Courses)
Elective- II

Sr. No.	Category	Course Code	Course Title
1	Discipline Specific Elective Courses	DSEC-MCA-115A	Theory of Computation
2		DSEC-MCA-117A	Advance Data Structures Using C++/Java

Table No. I (c) (Discipline Specific Elective Courses)
Elective- III

Sr. No.	Category	Course Code	Course Title
1	Discipline Specific Elective Courses	DSEC-MCA-112A	Software Engineering
2		DSEC-MCA-114A	Web Technologies

Table No. I (d) (Discipline Specific Elective Courses)
Elective- IV

Sr. No.	Category	Course Code	Course Title
1	Discipline Specific Elective Courses	DSEC-MCA-116A	Computer Networks and Distributed System
2		DSEC-MCA-118A	Advance Computer Architecture and Quantum Computing

Table –III (Pool of Foundation Elective Course defined by the Institute)
Foundation Elective Courses

SN	Course Code	Course Name	Offered by Department
1	FEC-158A	Basics of Accounting	Management Department
2	FEC-160A	Basics of E-commerce	Management Department
3	FEC-162A	Elements of Banking	Management Department
4	FEC-164A	Computer Fundamentals	Computer Science and Engineering
5	FEC-168A	Communication and Soft Skills	Applied Science and Humanities (English)
6	FEC-170A	Entrepreneurship Development	Management Department
7	FEC-172A	Electronics Engineering	Electronics and Communication Engineering

Course Code	**DSC-MCA-101A**				
Category	Discipline- Specific Courses				
Course Title	**Object Oriented Programming Using JAVA**				
Scheme and Credits	L	T	P	**Credits**	**Semester-I**
	4	0	0	**4**	
Course Objectives	The objectives of this course are to • Understand and apply Java basics, including syntax, data types, operators, control structures, and object-oriented principles to write simple Java programs. • Implement Java's object-oriented features, such as classes, inheritance, interfaces, and packages, to develop modular and reusable code. • Apply advanced Java concepts like exception handling, multithreading, and file I/O to build robust and concurrent applications. • Design and develop interactive GUI applications using Java applets, AWT, and graphics components.				
Assessment	40 Marks				
End Semester Examination	60 Marks				
Total Marks	100				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	**Skills Demonstrated**	**RBT Level**
CO1	Describe Java language fundamentals, program structure, inheritance, exception handling, and the applet lifecycle.	Level 1: Remember
CO2	Explain Java's class-based structure, string manipulation, exception handling, multithreading, and applet functionality.	Level 2: Understand
CO3	Develop Java programs using object-oriented principles, packages, interfaces, and multithreaded techniques for real-world problems.	Level 3: Apply
CO4	Analyze class hierarchies, exception-handling mechanisms, and thread synchronization to optimize Java applications.	Level 4: Analyze

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 8 parts of 1.5 marks each from all units and remaining eight questions of 12 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit-I

Introduction: Genesis and Evolution of Java Language, Internet & Java, Byte-code, its Features, Java Program Structure and Java's Class Library, Data Types, Variables, and Operators, Operator Precedence; Selection Statements, Scope of Variable, Iterative Statement; Defining Classes & Methods, Creating Objects of a Class, Defining and Using a Class, Automatic Garbage Collection.

Arrays and Strings: Arrays, Arrays of Characters, String Handling Using String Class, Operations on String Handling Using String Buffer Class.

Unit-II

Classes and Inheritance: Using Existing Classes, Class Inheritance, Choosing Base Class, Multiple Levels of Inheritance, Abstraction through Abstract Classes, Using Final Modifier. Packages: Understanding Packages, defining a Package, packaging up Your Classes, Adding Classes from a Package to Your Program, Understanding CLASSPATH, Standard Packages, Access Protection in Packages.

Interface Fundamentals: Creating an Interface, implementing an Interface, Using Interface References, Implementing Multiple Interfaces, Constants in Interfaces, Interfaces can be extended, Nested Interfaces, Final Thoughts on Interfaces.

Unit-III

Exception Handling: The concept of Exceptions, Types of Exceptions, dealing with Exceptions, Exception Objects, Defining Your Own Exceptions.

Multithreading Programming: The Java Thread Model, Understanding Threads, The Main Thread, creating a Thread, Creating Multiple Threads, Thread Priorities, Synchronization. I

Input/output in Java: I/O Basic, Byte and Character Structures, I/O Classes, Reading Console Input Writing Console Output, Reading and Writing on Files, Random Access Files, Storing and Retrieving Objects from File, Stream Benefits

Unit-IV

Applets in Java: Applet Basics, Applet Architecture, Applet Life Cycle, Simple Applet Display Methods, The HTML APPLET Tag Passing Parameters to Applets.

Working with Windows: AWT Classes, Window Fundamentals, working with Frame, creating a Frame Window in an Applet; Displaying Information within a Window.

Working with Graphics and Text: Working with Graphics, working with Color, Setting the Paint Mode, working with Fonts, Managing Text Output; Using Font Metrics, Exploring Text and Graphics, Working with AWT Controls, Layout Managers and Menus.

Suggested Readings:

- The Complete Reference JAVA, TMH Publication, July 2017.
- Beginning JAVA, Ivor Horton, WROX Public, 7 October 2011.
- JAVA 2 UNLEASHED, Tech Media Publications, 23 July2022.
- JAVA 2(1.3) API Documentations.
- Any other book(s) covering the contents of the paper in more depth.

Useful Video links:

Unit No.	Topics	Links
Unit-I	Genesis and Evolution of Java Language, Internet & Java, Byte-code, its Features	https://archive.nptel.ac.in/courses/106/105/106105191/#
	Java Program Structure and Java's Class Library, Data Types, Variables	https://www.youtube.com/watch?v=1B5ppTif5ZY&list=PLbRMhDVUMngcx5xHChJ-f7ofxZI4JzuQR&index=4
Unit-II	Class Inheritance, Choosing Base Class, Multiple Levels of Inheritance	https://archive.nptel.ac.in/courses/106/105/106105151/
	Using Final Modifier. Packages	https://www.youtube.com/watch?v=V9t_Ar1nHV4
Unit-III	Exception Handling	https://www.youtube.com/watch?v=iTGH6qSCy2I
	The Java Thread Model	https://www.youtube.com/watch?v=MNoPTVrsXZs
Unit-IV	Applets in Java	https://nptel.ac.in/courses/106105191
	AWT Classes	https://www.youtube.com/watch?v=4La3ChRGCtY

Course Code	**DSC-MCA-103A**				
Category	Discipline- Specific Courses				
Course Title	**Compiler Design**				
Scheme and Credits	L	T	P	**Credits**	**Semester-I**
	4	0	0	**4**	
Course Objectives	The objectives of this course are to • Understand system programming fundamentals, including assemblers, loaders, linkers, macros, and software tools. • Study compiler phases with a focus on lexical analysis, syntax analysis, and parsing techniques. • Learn intermediate code generation, syntax-directed definitions, and symbol table management. • Apply code optimization techniques and understand code generation principles like register allocation and global data flow analysis. • Develop system programs in C for lexical analysis, symbol table generation, and memory storage evaluation.				
Assessment	40 Marks				
End Semester Examination	60 Marks				
Total Marks	100 Marks				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	**Skills Demonstrated**	**RBT Level**
CO1	Describe fundamental concepts of system programming, compiler phases, and code optimization techniques.	Level 1: Remember
CO2	Explain the workings of loaders, lexical analyzers, parsers, symbol tables, and intermediate code generation processes.	Level 2: Understand
CO3	Implement system programs and compiler components such as tokenizes, parsers, and memory evaluators using programming languages like C.	Level 3: Apply
CO4	Analyze compiler design processes, optimization techniques, and the structure of syntax trees and symbol tables.	Level 4: Analyze

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 8 parts of 1.5 marks each from all units and remaining eight questions of 12 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit-I

Evolution of Systems Programming: Introduction to System programs, Overview of Assemblers, Loaders, Linkers, Macros, and Compilers. System Software Tools: Variety of software tools, Text editors, Interpreters and program generators, Debug Monitor, System Programming environment. Loader Schemes: Compile and Go Loader, general loader schemes, Absolute Loader, Subroutine linkage, Reallocating Loader, Direct Linkage Loader, Binders, Linking loader, overlays.

Unit-II

Compiler: Phases of Compiler, Compiler writing tools, Lexical Analysis, Finite Automata, Regular Expression, From a Regular expression to an NFA, NFA to DFA, Design of Lexical Analyzer. Syntax Analyzer, CFG, Role of the Parser, CFG, Top Down Parsing, Recursive Descent parsing, predictive Parsers, Bottom up Parsing, Shift reduce, Operator Precedence parsers, LR Parsers.

Unit-III

Intermediate Code: Syntax directed definitions, Evaluation Orders of Syntax directed definitions; Intermediate Languages: Intermediate code generation, Syntax trees, Construction of Syntax trees, Three Address Code, Types and Declarations, Translation of Expressions, Type Checking, Postfix form. Symbol table: Contents of Symbol table, Data Structures for Symbol table; Runtime Storage Administration.

Code Optimization and Code Generation: Principal sources of optimization, loop optimization, DAG - Optimization of Basic Blocks, Global Data Flow Analysis – Efficient Data Flow Algorithm. Code Generation: Issues in code generation, Design of a simple Code Generator, Register allocation and Assignment, Peephole optimization.

Unit-IV

System & Compiler programming: Developing system programs using C for basic OS commands apart from developing programs for lexical analysis, token counts, symbol table generator, memory storage requirement evaluator for identifiers for one or multiple declarative statements.

Suggested Readings:

- Donovan: Systems Programming, Tata McGraw Hill, 2009.
- Dhamdhere: System Software, Tata McGraw Hill,July 2011.
- Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman: Compilers Principles, Techniques and Tools, Addison Wesley, 2017.
- Alfred V.Aho and Jeffrey D.Ullman: Principles of Compiler Design, Addison Wesley, January 2002.
- William M. Waite, Gerhard Goos: Compiler Construction, May 2013.
- TorbenÆgidius Mogensen: Basics of Compiler Design, ISBN 978-87-993154-0-6.
- Bergmann, Seth D: Compiler Design: Theory, Tools and Examples, 2017.
- Any other book(s) covering the contents of the paper in more depth.

Useful Video links:

Unit No.	Topics	Links
Unit-I	Overview of Assemblers, Loaders, Linkers, Macros, Compilers	https://www.nptelvideos.com/lecture.php?id=5232
Unit-II	Compiler Phases	https://www.nptelvideos.com/lecture.php?id=5234
	Lexical Analysis	https://www.nptelvideos.com/lecture.php?id=5235
	Shift reduce	https://www.nptelvideos.com/lecture.php?id=5240
Unit-III	Intermediate code generation	https://www.nptelvideos.com/lecture.php?id=5234
	Postfix form. Symbol table	https://www.nptelvideos.com/lecture.php?id=5246
Unit-IV	Global Data Flow Analysis	https://www.nptelvideos.com/lecture.php?id=5254
	Register allocation and Assignment	https://nptelvideos.com/lecture.php?id=5258

Course Code	**DSC-MCA-105A**				
Category	Discipline- Specific Courses				
Course Title	**Computer Graphics and Multimedia**				
Scheme and Credits	L	T	P	**Credits**	**Semester-I**
	4	0	0	**4**	
Course Objectives	The objectives of this course are to • Understand computer graphics fundamentals, including display devices, graphics primitives, and scan conversion algorithms. • Apply 2D and 3D transformation techniques like translation, scaling, rotation, and clipping. • Analyze and implement 3D object representation methods, hidden surface detection, and shading techniques. • Understand multimedia systems, including architecture, compression, authoring, and data standards for integrated content creation.				
Assessment	40 Marks				
End Semester Examination	60 Marks				
Total Marks	100 Marks				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	**Skills Demonstrated**	**RBT Level**
CO1	Describe fundamental concepts of computer graphics, multimedia systems, color models, and 2D/3D transformations.	Level 1: Remember
CO2	Explain scan conversion algorithms, 3D object representations, multimedia components, and compression-decompression techniques.	Level 2: Understand
CO3	Apply basic graphical primitives, transformations, and multimedia projects using tools like FLASH or BLENDER.	Level 3: Apply
CO4	Analyze viewing pipelines, clipping algorithms, surface detection techniques, and multimedia system architectures.	Level 4: Analyze

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 8 parts of 1.5 marks each from all units and remaining eight questions of 12 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit-I

Basics of Computer Graphics: Computer Graphics, Classification, Applications of computer graphics, Display devices, Random and Raster scan systems, Graphics input devices, Graphics software and standards.

Graphics Primitives: Points, lines, circles and ellipses as primitives, scan conversion algorithms for primitives, Fill area primitives including scan-line polygon filling, inside- outside test, boundary and flood-fill, character generation, line attributes, area-fill attributes, character attributers.

Unit-II

2D Transformation and Viewing: Transformations (translation, rotation, scaling), matrix representation, homogeneous coordinates, composite transformations, reflection and shearing, viewing pipeline and coordinates system, window-to-viewport transformation, clipping including point clipping, line clipping (cohen-sutherland, liang- bersky, NLN), polygon clipping.

3D Concepts and Object Representation: 3D display methods, polygon surfaces, tables, equations, meshes, curved lies and surfaces, quadric surfaces, spline representation, cubic spline interpolation methods, Bazier curves and surfaces, B-spline curves and surfaces.

Unit-III

3D Transformation and Viewing: 3D scaling, rotation and translation, composite transformation, viewing pipeline and coordinates, parallel and perspective transformation, view volume and general (parallel and perspective) projection transformations. Modelling: Wireframe and Solid.

Hidden Surfaces: Visible surface detection concepts, Back-face detection, Depth Buffer method, Illumination, Light sources, Illumination methods (ambient, diffuse reflection, specular reflection). Color models: properties of light, XYZ, RGB, YIQ and CMY color models. Shading: Flat, Gouraud and Phong.

Unit-IV

Multimedia Basics: Concepts of Multimedia, Multimedia applications, Multimedia system architecture, Evolving technologies for multimedia, Defining objects for multimedia systems, Multimedia data interface standards, Multimedia databases. Compression and decompression: Data and file format standards, Multimedia I/O technologies, Digital voice and audio, Video image and animation, Full motion video, Storage and retrieval technologies.

Multimedia Authoring: Concept of Multimedia Authoring, Hypermedia messaging, Mobile messaging, Hypermedia message component, Creating hypermedia message, Integrated multimedia message standards, Integrated document management, Distributed multimedia systems.

Case Study (FLASH/ BLENDER): Drawing Basic Shapes, Modeling, Shading & Textures, Creating a multimedia project.

Suggested Readings:

- Donald Hearn and M.Pauline Baker: Computer Graphics, PHI Publications, 1986
- Plastock: Theory & Problem of Computer Graphics, Schaum Series, 1986.
- Foley & Van Dam: Fundamentals of Interactive Computer Graphics, Addison-Wesley, 1982.
- Newman: Principles of Interactive Computer Graphics, McGraw Hill, 1979.
- Tosijasu, L.K.: Computer Graphics, Springer-Verlag, 1991.
- S Gokul: Multimedia Magic, BPB Publication, 2000.
- Bufford: Multimedia Systems, Addison Wesley, 1996.
- Jeffcoate : Multimedia in Practice, Prentice-Hall,1995.

Useful Video links:

Unit No.	Topics	Links
Unit-I	Computer Graphics	https://www.nptelvideos.com/lecture.php?id=6261
	Classification of Computer Graphics	https://www.nptelvideos.com/lecture.php?id=6262
	Display Devices	https://www.nptelvideos.com/lecture.php?id=6263
Unit-II	clipping including point clipping, line clipping (cohen-sutherland, liang- bersky, NLN)	https://www.nptelvideos.com/lecture.php?id=6264
	Transformations (translation, rotation, scaling)	https://www.nptelvideos.com/lecture.php?id=6266
	polygon surfaces	https://www.nptelvideos.com/lecture.php?id=6276
Unit-III	Parallel and Perspective Transformation	https://www.nptelvideos.com/lecture.php?id=6268
	Viewing Pipeline for 3D	https://www.nptelvideos.com/lecture.php?id=6288
Unit-IV	Video image and animation	https://www.nptelvideos.com/lecture.php?id=6293
	Hypermedia message component	https://nptel.ac.in/courses/106106090

Course Code	**DSEC-MCA-111A**				
Category	Discipline Specific Elective Courses				
Course Title	**Digital Design and Computer Architecture**				
Scheme and Credits	L	T	P	**Credits**	**Semester-I**
	4	0	0	**4**	
Course Objectives	The objectives of this course are to • Understand number systems, binary arithmetic, and Boolean algebra for logic simplification. • Design and analyze combinational and sequential circuits, including adders, multiplexers, flip-flops, and counters. • Apply computer architecture concepts like instruction sets, CPU design, and assembly language programming. • Understand I/O organization, memory management, and advanced architectures (parallel processing, pipelining, SIMD) for real-world applications.				
Assessment	40 Marks				
End Semester Examination	60 Marks				
Total Marks	100 Marks				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	**Skills Demonstrated**	**RBT Level**
CO1	Describe number systems, binary arithmetic, Boolean algebra, and basic instruction sets in computer architecture.	Level 1: Remember
CO2	Explain combinational and sequential circuits, CPU micro-operations, addressing modes, and pipelining principles in computer systems.	Level 2: Understand
CO3	Implement basic digital circuits, registers, counters, and assembly language programs for 8086/8088 processors.	Level 3: Apply
CO4	Analyze control unit designs, instruction cycles, and the functioning of parallel processing architectures for improved efficiency.	Level 4: Analyze

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 8 parts of 1.5 marks each from all units and remaining eight questions of 12 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit-I

Number System: Binary, Octal, Hexadecimal and Decimal, 1's and 2's Complements, Inter conversion of numbers. Codes: Weighted and Non-weighted codes, BCD Codes, Gray codes, Self-complementing codes, Error-Detecting/Correcting codes, Alphanumeric Codes, Hamming Codes, Floating Point Numbers.
Binary Arithmetic: Binary Addition and Subtraction, 2's Complement Arithmetic, Booth Coding, Binary Multiplication.

Logic Design: Logic Gates, Truth Tables, Boolean Algebra, Boolean Expressions-Variables and Literals, Boolean Expressions–Equivalent and Complement, Theorems of Boolean Algebra, Simplification Techniques, SOPs & POSs Boolean Expressions.

Unit-II

Combinational Circuits: Combinational Logic, Arithmetic Circuits– Adder and Subtractor, BCD Adder, Code Converters, Magnitude Comparator, Parity Generators/Checkers, Multiplexers, Demultiplexers, Decoders, Encoders.

Sequential Circuits: Latches, R S Flip Flop, Level Triggered and Edge Triggered Flip Flops, JK Flip-Flop, Master-Slave Flip Flops, T Flip-Flop, D Flip-Flops.

Registers and Counters: Controlled Buffer Registers, Shift Registers, Applications of Shiftregisters; Ripple Counter, Synchronous Counter, Modulus Counter, Binary Ripple Counters, Up/Down Counters, Decade and BCD Counters.

Unit-III

Basic Computer Design: Computer Instructions and types, Instruction Set, Instruction Cycle, Instruction Formats, Addressing Modes, Computer Registers, Bus System, Register Transfer Language terminology.

Programming in 8086/8088 Assembly Language: A/L program structure, segments, registers, instructions, macros, A/L directives.

CPU Design: CPU Registers, Micro-operations and its types, Design of ALU. Control Unit Design-Microprograms, Control Unit of a basic computer–Timing and Control; Hardwired and Micro-programmed controlled unit. Architectures -RISC, CISC, Scalar, Superscalar and pipelined architectures.

Unit-IV

Input/output Organization: Peripheral Devices, Input-output Interface, Asynchronous Data Transfer, Mode of Transfer, Priority Interrupt, Direct Memory Access, Input-output Processor, Serial Communication.

Advance Architecture: Introduction to parallel processing– Pipelining, Parallel Computer structures, Architectural classification. Pipelining & Vector processing; Instruction and Arithmetic pipelines, Principles of designing pipelined processors, Structures for array processors: SIMD Array processor, SIMD Interconnection networks. Parallel Processing Applications a multimedia project.

Suggested Readings:

- Mano, M.M.: Digital Logic and Computer Design, Prentice-Hall of India, 1979 (First Edition).
- Gill Nasib Singh and Dixit J.B: Digital Design and Computer Organisation, University Science Press (Laxmi Publications), New Delhi, 2009 (First Edition).
- Stallings, William: Computer Organisation & Architecture, 2020 (10th Edition).
- Mano, M.M.: Digital Design, Prentice-Hall of India, 2001 (4th Edition).
- Anand Kumar: Fundamentals of Digital Circuits, PHI, 2008 (2nd Edition).
- Kai Hwang: Advanced Computer Architecture, McGraw Hill International, 1993 (First Edition)
- Mano,M.M.: Computer System Architecture, Prentice-Hall of India, 2007 (3rd Edition).
- Tokheim: Digital Electronics, TMH, 1999 (5th Edition).
- Any other book(s) covering the contents of the paper in more depth.

Useful Video links:

Unit No.	Topics	Links
Unit-I	Binary, Octal, Hexadecimal, Decimal Conversions	https://www.nptelvideos.com/lecture.php?id=5269
Unit-II	Multiplexers and Decoders	https://www.nptelvideos.com/lecture.php?id=5836
	Error-Detecting/Correcting codes	https://www.nptelvideos.com/lecture.php?id=5841
Unit-III	Computer Instructions and types	https://archive.nptel.ac.in/courses/106/105/106105163/
	pipelined architectures	https://archive.nptel.ac.in/courses/117/105/117105078/
Unit-IV	Input-output Interface	https://www.youtube.com/watch?v=09jCRiVWsJw&t=2s
	Introduction to parallel processing	https://www.youtube.com/watch?v=NypfUUd6M8k

Course Code	**DSEC-MCA-117A**				
Category	Discipline Specific Elective Courses				
Course Title	**Advance Data Structures Using C++/JAVA**				
Scheme and Credits	L	T	P	**Credits**	**Semester-I**
	4	0	0	**4**	
Course Objectives	The objectives of this course are to • Understand algorithm analysis, including time and space complexity, asymptotic notations, and strategies for efficient algorithm design. • Implement essential data structures, such as trees and heaps, and applying traversal methods for effective data organization and management. • Introduce graph algorithms, such as shortest path, minimum spanning tree, and network flow algorithms, to solve complex computational problems. • Introduce concepts of computational complexity, such as NP-completeness, for classifying and approaching hard problems.				
Assessment	40 Marks				
End Semester Examination	60 Marks				
Total Marks	100 Marks				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	**Skills Demonstrated**	**RBT Level**
CO1	Describe key algorithms, graph representations, tree traversal methods, dynamic programming techniques, and complexity classes in algorithm design.	Level 1: Remember
CO2	Explain the principles of algorithm analysis, tree and graph algorithms, dynamic programming, and NP-completeness in computational theory.	Level 2: Understand
CO3	Implement and execute various algorithms, such as divide-and-conquer, backtracking, graph algorithms, and string-matching, using C++/Java.	Level 3: Apply
CO4	Analyze the time and space complexity of algorithms using asymptotic notations and evaluate the efficiency of data structures in problem-solving.	Level 4: Analyze

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 8 parts of 1.5 marks each from all units and remaining eight questions of 12 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit-I

The Role of Algorithms in computing: Analyzing Algorithms, Time and Space Analysis of Algorithms, Big-Oh and Theta Notations, Average, Best and Worst case analysis. Designing Algorithms, Growth of functions. Asymptotic Notations, Divide and Conquer, Recurrences, Maximum sub-array problem, Stressan's Method, Substitution method, Recurrence tree method, The Master method, Floors and Ceilings.

Unit-II

Trees: Binary tree traversal methods: Pre-order, In-order, Post-ordered traversal. Recursive Algorithms. Traversal methods. Representation of trees and its applications: Binary tree representation of a general tree. Conversion of forest into tree. Threaded binary trees. Binary search tree: Height balanced (AVL) tree, B-trees, Splay tree. Heap: Heap operations, Binomial heaps, Fibonacci heaps, Skew heaps, heap set.

Unit-III

Graphs & Algorithms: Representation, Type of Graphs, Paths and Circuits: Euler Graphs, Hamiltonian Paths & Circuits; Cut-sets, Connectivity and Separability, Planar Graphs, Isomorphism, Graph Coloring, Covering and Partitioning, , Depth-and breadth-first traversals, Minimum Spanning Tree: Prim's and Kruskal's algorithms, Shortest-path Algorithms: Dijkstra's and Floyd's algorithm, Topological sort, Maxflow: Ford-Fulkerson algorithm, max flow –min cut.

Unit-IV

Dynamic Programming: Backtracking Algorithms, Design Methodologies, Travelling salesperson problem, 0/1 Knapsack problem, multistage graphs, All Pair Shortest Path, 8-Queens problem Advanced String-Matching Algorithms: Naïve string-matching algorithm, Robin-Karp algorithm, string matching with finite automata, Knuth-Morris-Pratt algorithm.

P, NP and Approximation Algorithms: Basic Concepts, Non Deterministic algorithms, NP Complete and NP-hard classes, NP complete Problems.

Implementation of above mentioned data structures & algorithms through C++/Java programming.

Suggested Readings:

- Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest: Introduction to Algorithms, PHI Learning Pvt. Ltd, 2009.
- Gilles Brassard, Paul Bratley: Fundamentals of Algorithms, PHI Learning Pvt. Ltd, 2011.
- Hubbard JR: Schaum's Outline of Data Structures with C++, Tata McGraw Hills, New Delhi, 2000.
- R. Sedgewick: Algorithms in C++, Pearson Education Asia, 2001.
- Y. Langsam, M.J.Augenstein and A.M.Tanenbaum: Data Structures Using C and C++, Prentice Hall of India, 1996.
- R.Kruse, C.L.Tonodo and B.Leung: Data Structures and Program Design in C, Pearson Education. New Delhi, 1997.
- G.L. Heileman: Data Structutes: Algorithms and Object-Oriented Programming, Tata McGraw Hill, New Delhi, 2009.
- E. Horowitz, Sahni and D. Mehta: Fundamentals of Data Structures in C++, Galgotia Publication, New Delhi, 2008.

Useful Video links:

Unit No.	Topics	Links
Unit-I	Analyzing Algorithms	https://www.nptelvideos.com/lecture.php?id=5947
	Designing Algorithms	https://www.nptelvideos.com/lecture.php?id=5950
Unit-II	Binary tree traversal methods	https://www.youtube.com/watch?v=hV4AFZS81wU
	Height balanced (AVL) tree	https://www.youtube.com/watch?v=YOtXKE1Kjdk
Unit-III	Dijkstra's and Floyd's algorithm	https://archive.nptel.ac.in/courses/106/106/106106127/
	Prim's and Kruskal's algorithms	https://nptel.ac.in/courses/106102064
Unit-IV	NP Complete and NP-hard classes	https://www.nptelvideos.com/lecture.php?id=5971

<table>
<tr><td>Course Code</td><td colspan="5">EL-MCA-101A</td></tr>
<tr><td>Category</td><td colspan="5">Experiential Learning</td></tr>
<tr><td>Course Title</td><td colspan="5">Software Lab-1</td></tr>
<tr><td rowspan="2">Scheme and Credits</td><td>L</td><td>T</td><td>P</td><td>Credits</td><td rowspan="2">Semester-I</td></tr>
<tr><td>0</td><td>0</td><td>6</td><td>3</td></tr>
<tr><td>Course Objectives</td><td colspan="5">The objectives of this course are to
Apply Java syntax, data structures, control flow, and OOP principles (classes, objects, inheritance, and polymorphism).
Design and implement Java apps using OOP features, exception handling, and multithreading for efficient software.
Design user-friendly GUIs using AWT components (buttons, labels, text fields) for smooth user interactions.
Manage errors with exception propagation, exception objects, and effective exception handling strategies.</td></tr>
<tr><td>Internal Assessment</td><td colspan="5">50Marks</td></tr>
<tr><td>External Practical</td><td colspan="5">50 Marks</td></tr>
<tr><td>Total</td><td colspan="5">100Marks</td></tr>
<tr><td>Duration of Exam</td><td colspan="5">03 Hours</td></tr>
</table>

Course Outcomes: After studying this course, the students will be able to

COs	**Skills Demonstrated**	**RBT Level**
CO1	Implement object-oriented concepts such as inheritance, abstraction, interfaces, event handling, and graphics algorithms in Java and C/C++.	Level 3: Apply
CO2	Analyze and debug Java programs, algorithms for geometric transformations, and multi-threading applications for optimization and error resolution.	Level 4: Analyze
CO3	Evaluate the efficiency and functionality of matrix multiplication, event handling techniques, graphical algorithms, and multi-threading in Java and C/C++.	Level 5: Evaluate
CO4	Design and develop Java applications and C/C++ programs for graphical transformations, event-driven programming, and multi-threading with real-time functionality.	Level 6: Create

List of Experiments:

Sr. No.	Contents
1	Write a java program to multiply two given matrices
2	Write a java program for Method overloading and Constructor overloading
3	Write a java program to display the employee details using Scanner class
4	Write a java program to represent Abstract class with example.
5	Write a java program to implement Interface using extends keyword
6	Write a Java program that implements a multi-thread application that has three threads
7	Write an applet program that displays a simple message
8	Write a java program for handling Mouse events and Key events
9	Write a java program for handling Key events
10	Write a java program to find the Fibonacci series using recursive and non-recursive functions
11	Write a java program that checks whether a given string is palindrome or not
12	Write a java program to create inner classes.

Suggested Readings:

- The Complete Reference JAVA, TMH Publication, 1972.
- JAVA 2 UNLEASHED, Tech Media Publications, 2002.

Useful Video Links:

Experiment No.	Experiment Name	Links
1	Introduction of java and types	https://archive.nptel.ac.in/courses/106/105/106105191/
2	Arrays and Strings types in java	https://www.youtube.com/watch?v=VksxhzfD8kQ
3	Packages in java	https://www.youtube.com/watch?v=TwU3cv1FFis
4	Interface Fundamentals in java	https://www.youtube.com/watch?v=nFH-Hm8iTR0

Course Code	**EL-MCA-117A**				
Category	Experiential Learning				
Course Title	**Software Lab-2**				
Scheme and Credits	L	T	P	**Credits**	**Semester-I**
	0	0	6	**3**	
Course Objectives	The objectives of this course are to • Understand and Apply Core Concepts of Algorithm Analysis. • Design and Implement Tree Data Structures and Traversal Algorithms. • Implement Graph Algorithms for Problem Solving. • Solve Optimization Problems Using Dynamic Programming and Backtracking. • Implement Advanced String-Matching Algorithms for Text Searching.				
Internal Assessment	50 Marks				
External Practical	50 Marks				
Total	100Marks				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	**Skills Demonstrated**	**RBT Level**
CO1	Apply sorting algorithms, tree traversals, graph search techniques, and string matching algorithms to solve computational problems in programming.	Level 3: Apply
CO2	Analyze the time and space complexities of sorting algorithms, tree operations, and optimization problems, evaluating their efficiency.	Level 4: Analyze
CO3	Critically assess the performance and correctness of algorithms such as KMP, Prim's, Kruskal's, and dynamic programming solutions for optimization problems.	Level 5: Evaluate
CO4	Design and develop efficient solutions for graph and tree-based problems, including tree balancing, minimum spanning tree, and optimization problems.	Level 6: Create

List of Experiments

Sr. No.	Contents
1	Implementing and Analyzing Sorting Algorithms: Bubble Sort and Selection Sort
2	Analyzing the Time and Space Complexity of Binary Search
3	Solving Recurrences Using the Master Theorem
4	Binary Tree Traversal: Pre-order, In-order, and post-order
5	Implementing Binary Search Tree (BST) Operations: Insert, Delete, Search
6	Implementing AVL Tree for Balancing a Binary Search Tree
7	Implementing Depth-First Search (DFS) and Breadth-First Search (BFS)
8	Finding Minimum Spanning Tree Using Prim's Algorithm
9	Finding Minimum Spanning Tree Using Kruskal's Algorithm
10	Solving the 0/1 Knapsack Problem Using Dynamic Programming
11	Solving the Travelling Salesman Problem Using Backtracking
12	Solving the N-Queens Problem Using Backtracking
13	Naïve String-Matching Algorithm
14	Implementing Knuth-Morris-Pratt (KMP) String Matching Algorithm

Suggested Readings:

1. Gilles Brassard, Paul Bratley: Fundamentals of Algorithms, PHI Learning Pvt. Ltd, 2011
2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest: Introduction to Algorithms, PHI Learning Pvt. Ltd,2009
3. Hubbard JR: Schaum's Outline of Data Structures with C++, Tata McGraw Hills, New Delhi, 2000.
4. Y.Langsam, M.J.Augenstein and A.M.Tanenbaum: Data Structures Using C and C++, Prentice Hall of India, 2015.

Useful Video Links:

Experiment No.	Experiment Name	Links
1	Sorting Algorithms: Bubble Sort and Selection Sort	https://archive.nptel.ac.in/courses/106/102/106102064/
2	Time and Space Complexity of Binary Search	https://www.youtube.com/watch?v=zWg7U0OEAoE
3	Prim's Algorithm	https://www.nptelvideos.com/lecture.php?id=5899
4	Depth-First Search (DFS)	https://www.nptelvideos.com/lecture.php?id=5893

Course Code	**BC-MCA-111A**				
Category	Bridge Course				
Course Title	**Computer Fundamental and Programming in C**				
Scheme and Credits	L	T	P	**Credits**	**SEMESTER – I**
	4	0	0	**4**	
Course Objectives	The objectives of this course are to • Understand computer basics and role of operating system. • Learn about concept of computer network, Internet and social impacts of IT. • Design an algorithm and draw flowchart for simple problems. • Develop C programs implementing all features of C.				
Assessment	40 Marks				
End Semester Examination	60 Marks				
Total Marks	100				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	**Skills Demonstrated**	**RBT Level**
CO1	Describe fundamental computer concepts, programming basics, and network terminologies, including data, operating systems, and file handling.	Level 1: Remember
CO2	Explain the working of computers, application software, and network protocols, emphasizing problem-solving techniques and IT impacts.	Level 2: Understand
CO3	Use programming constructs, office tools, and file management techniques to create and manage efficient solutions for real-world tasks.	Level 3: Apply
CO4	Break down complex problems using algorithms, flowcharts, and pseudo code, analyzing programming logic and network topologies for optimization.	Level 4: Analyze

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 8 parts of 1.5 marks each from all units and remaining eight questions of 12 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit-I

Computer Fundamentals: Concept of data and information, Historical evolution of computers, Block Diagram of Computer and working, Characteristics, Classification of Computers, Advantages and Limitations of Computer, Applications of Computer, I/O Devices, Memory and Storage Devices;

Computer Software: System and Application Software.

Operating System: Characteristics, bootstrapping, types of Operating System, Operating System as resource manager.

Programming Languages: Machine, Assembly, High Level Language, 4GL. Language Translator, System Utilities- Editor, Linker, Loader, File Manager.

Computer Network Concepts: Definition, Types of Networks, Topology, Protocols, Intranet, Extranet, Internet, WWW, Search Engine, Web Browsers, Services of Internet. IT and Social Impacts of IT: Positive and Negative Impacts, Computer Crimes, Viruses and their remedial solutions.

Unit-II

Word Processor: Introduction, Windows Interface, Customizing the Word Application, Document Views, Basic Formatting in MS Word, Advanced Formatting, navigating through a Word Document, Performing a Mail Merge, A Quick Look at Macros, Printing Documents, Print Preview

Spreadsheet Application: Introduction, Workbook, Worksheet, formatting in excel, Advanced formatting in Excel, Working with formulas, Printing worksheets

Presentation Software: Introduction, creating a Presentation, Basic Formatting in PowerPoint, Advanced Formatting, Using Templates, inserting charts, Inserting tables, Printing presentations.

Unit-III

Problem Solving: Problem Identification, Analysis, Algorithms, Flowcharts, Pseudo codes, Decision Tables, Program Coding, Program Testing and Execution.

C Programming Fundamentals: Basic Concepts, Structure of a C program, Operators & Expressions; Library Functions, Decision making using if…else, Else If Ladder; Switch, break, Continue and Goto statements, Control Statements: Looping using while, do…while, for statements, Nested loops.

Arrays & Functions: Declaration and Initialization, Multidimensional Arrays, String: Operations of Strings, Functions: Defining & Accessing User defined functions, Function Prototype, Passing Arguments, Passing array as argument, Recursion, Use of Library Functions, Macro vs. Functions.

Unit-IV

Pointers: Declarations, Operations on Pointers, passing to a function, Pointers & Arrays, Array of Pointers, Array accessing through pointers, Pointer to functions, Function returning pointers, Dynamic Memory Allocations.

Structures and Union: Defining and Initializing Structure, Array within Structure, Array of Structure, Nesting of Structure, Pointer to Structure, Passing structure and its pointer to Functions, Unions: Introduction to Unions and its Utilities.

File Handling: Opening and closing file in C, Create, Read and Write data to a file, Modes of Files, Operations on file using C Library Functions, Working with Command Line Arguments, Program Debugging and types of errors.

Suggested Readings:

- Gill Nasib Singh: Computing Fundamentals and Programming in C, Khanna Books Publishing Co., New Delhi, 30 December 2015.
- Kenneth. A.: C problem solving and programming, Prentice Hall, 1990.
- Gottfried, B.: Theory and problems of Programming in C, Schaum Series, 1996.
- Gill, Nasib Singh: Handbook of Computers, Khanna Books Publishing Co., New Delhi, 2016.
- Sanders, D.: Computers Today, Tata McGraw-Hill, 1988.
- Cooper, Mullish: The spirit of C, An Introduction to Modern Programming, Jaico Publ. House, New Delhi, 2016.
- Kernighan & Ritchie: The C Programming Language, PHI, 1988.

Useful Video links:

Unit No.	Topics	Links
Unit-I	Operating System	https://archive.nptel.ac.in/courses/106/105/106105214/
	Topology	https://archive.nptel.ac.in/courses/106/105/106105081/
Unit-II	Advanced formatting in Excel	https://www.youtube.com/watch?v=uisSkBOGlUM
Unit-III	Basic Concepts and Structure of a C program	https://archive.nptel.ac.in/courses/106/104/106104128/
	Decision making using if…else, Else If Ladder	https://archive.nptel.ac.in/courses/106/105/106105171/
Unit-IV	Operations on Pointers	https://www.youtube.com/watch?v=Z_0xXmOgYtY
	File Handling	https://archive.nptel.ac.in/courses/106/104/106104128/

Course Code	**BC-MCA-121A**				
Category	Bridge Course				
Course Title	**C++ and Data Structures**				
Scheme and Credits	L	T	P	**Credits**	**SEMESTER-I**
	4	0	0	**4**	
Course Objectives	The objectives of this course are to • Understand concept of object oriented programming and its features • Understand the importance of class design in C++ • Understand importance of polymorphism and inheritance • Implement key data structures like arrays, stacks, queues, and linked lists, and teach students how to represent and manipulate these structures in C++ for solving real-world computational problems.				
Assessment	40 Marks				
End Semester Examination	60 Marks				
Total Marks	100				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	**Skills Demonstrated**	**RBT Level**
CO1	Describe core object-oriented programming principles, algorithm concepts, data structures, and memory management techniques.	Level 1: Remember
CO2	Explain OOP characteristics, inheritance, polymorphism, data structures, and their real-world applications.	Level 2: Understand
CO3	Implement OOP concepts, algorithms, sorting techniques, and data structures to solve computational problems effectively.	Level 3: Apply
CO4	Examine algorithms, data structures, and OOP design principles to optimize time and space complexity in programs.	Level 4: Analyze

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 8 parts of 1.5 marks each from all units and remaining eight questions of 12 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit-I

Introduction to OOP: Concept of OOP, Procedural vs. Object oriented programming, Characteristics of OOP: Objects, classes, Encapsulation, Data Abstraction, Inheritance, Polymorphism, Dynamic Binding, and Message Passing.

C++ Programming: Data-types, Variables, Static Variables, Operators in C++, Arrays, Strings, Structure, Functions, Recursion, Control Statements.

Access Specifies: Private, Public and Protected, Member functions of the class, Constructor and Destructor, Parameterized Constructor, Copy Constructors.

Unit-II

Inheritance: Reusability, Types of Inheritance: Single inheritance, Multiple, Multilevel, Hybrid Inheritance, Public, Private, and Protected Derivations.

Polymorphism: Function Overloading, Static Class Members, Static Member Functions, Friend Functions. Operator

Overloading: Unary and Binary Operator Overloading, Abstract class, Virtual function, pure virtual function, Overloading vs. Overriding. Memory management: new, delete, object Creation at Run Time. Exception handling: Throwing, Catching, and Re-throwing an exception.

Unit-III

Design and Analysis of Algorithm: Algorithm definition, comparison of algorithms. Top down and bottom-up approaches to Algorithm design.

Introduction to Data Structures: Concept of Data Structure, Types of Data Structure: Primitive and non–primitive.

Arrays: Single and Multidimensional arrays. Address calculation using column and row major ordering. Various Operations on arrays. Applications of arrays.

Sorting: Selection sort, Insertion sort, Bubble sort, Quick sort, merge sort, Radix sort. Searching: Sequential and binary search, Indexed search, Hashing Schemes. Comparison of time complexity

Unit-IV

Stacks and Queues: Representation of stacks and queues using arrays and linked-list.

Applications of stacks: Conversion from infix to postfix and prefix expressions, Evaluation of postfix expression using stacks.

Linked list: Singly linked list; operations on list, Linked stacks and queues. Polynomial representation and manipulation using linked lists. Circular linked lists, doubly linked lists.

Applications of Stack, Queue and Linked List data structures.

Suggested Readings:

- Herbert Schildt: C++ - The Complete Reference, Tata McGraw Hill Publications, 2017.
- E. Balaguruswamy: C++, Tata McGraw Hill Publications, 2020.
- E. Balaguruswamy: Object Oriented Programming and C++, TMH, 2008.
- Shah &Thakker: Programming in C++, ISTE/EXCEL, 2002.
- Johnston: C++ Programming Today, PHI, 2007.

Useful Video links:

Unit No.	Topics	Links
Unit-I	C++ Programming	https://archive.nptel.ac.in/courses/106/105/106105234/#
	Classes, Functions and constructors	https://www.youtube.com/watch?v=6niBuqibHZk
Unit-II	Inheritance	https://archive.nptel.ac.in/courses/106/105/106105151/
	Polymorphism	https://www.youtube.com/watch?v=N0iYriJWq94&t=566s
Unit-III	Algorithm	https://www.youtube.com/watch?v=zWg7U0OEAoE&t=1s
	Array	https://www.youtube.com/watch?v=mjxdMDItUjo
Unit-IV	Stacks and Queues	https://www.youtube.com/watch?v=g1USSZVWDsY&t=1s
	Linked list	https://nptel.ac.in/courses/106102064

Course Code	**BC-MCA-123A**				
Category	Bridge Course				
Course Title	**Visual Basic and Database Systems**				
Scheme and Credits	L	T	P	**Credits**	**SEMESTER -I**
	4	0	0	**4**	
Course Objectives	The objectives of this course are to • Design user interfaces and develop basic applications using fundamental controls, variables, and control structures. • Implement menus and enhance application functionality in Visual Basic. • Introduce foundational concepts of Database Management Systems, including data models, ER modeling, relational models, and query languages. • Teach database normalization techniques, ACID properties, and concurrency control to optimize performance and ensure data integrity.				
Assessment	40 Marks				
End Semester Examination	60 Marks				
Total Marks	100				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	**Skills Demonstrated**	**RBT Level**
CO1	Describe fundamental concepts of Visual Basic, database management systems, ER modeling, and normalization techniques.	Level 1: Remember
CO2	Explain the principles of event-driven programming, database design, and concurrency control in database systems.	Level 2: Understand
CO3	Develop Visual Basic applications using forms, dialog boxes, menus, and data controls to interact with databases.	Level 3: Apply
CO4	Analyze database schemas for redundancy, apply normalization techniques, and evaluate transaction schedules for concurrency issues.	Level 4: Analyze

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 8 parts of 1.5 marks each from all units and remaining eight questions of 12 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit-I

Introduction to Visual Basic: VB IDE, An overview of VB project types, VB as event driven & object-based language, Default Controls in Tool Box: Label Box, Text Box, Command Button, List Box, Combo Box, Picture & Image Box, Shape box, Timer, Option button, Check Box & Frames. Exploring Project Properties.

Programming with VB: Variables, Constants, Data types, Variable Scope, Arithmetic operations, String Operations, Built-in functions, I/O in VB, Branching & Looping statements, Procedures, Arrays, Collection.

Unit-II

Working with Forms: Working with multiple forms; Loading, Showing and Hiding forms; Creating Forms at Run Time, Drag and Drop operation, MDI form, Arranging MDI Child Windows, Coordinating Data between MDI Child Forms.

Dialog Boxes and Menu: Using Common Dialog Box; Adding Menu, Modifying and Deleting Menu Items, Creating Submenus.

VB & Databases: The Data Controls and Data-Bound Controls; Using DAO, RDO, ADO

Unit-III

Database Management System: Introduction, Database System Applications, History of Database Systems, Database System Vs. File Processing System, View of Data, Data Abstraction, Instances and Schemas. DBMS Environment, Database languages, Database Models.

Database design and ER Model: Physical, Conceptual and Logical Database design, ER Modelling, Conceptual design with ER Model

Unit-IV

Normalization: Schema Refinement, Problems caused by redundancy, Decomposition & its properties; Normalization: FIRST, SECOND, THIRD Normal forms, BCNF, Multivalued Dependencies, Join Dependencies. Transaction Management.

Concurrency Control: ACID properties, Transactions and Schedules, Concurrent execution of transaction, Serializability and Recoverability, Lock based Concurrency control, Lock Management, Lock Conversion, Dealing with deadlocks, Concurrency without Locking

Suggested Readings:

- Steven Holzner: Visual Basic 6 Programming: Black Book, Dreamtech PRESS,1998.
- Evangelos Petroutsos: Mastering Visual Baisc 6, BPB, 1998.
- Julia Case Bradley & Anita C.: Mills paugh Programming in Visual Basic 6.0, Tata McGraw-Hill,1999.
- Michael Halvorson: Step by Step Microsoft Visual Basic 6.0 Professional, PHI, 2022.
- Scott Warner: Teach Yourself Visual basic 6, Tata McGraw-Hill Edition, 1998.
- Elmasri & Navathe: Fundamentals of Database Systems, 5th edition, Pearson Education,2018.
- Thomas Connolly, Carolyn Begg: Database Systems, Pearson Education,2019.
- C. J. Date: An Introduction to Database Systems, 8th edition, Addison Wesley N. Delhi, 2006.

Useful Video links:

Unit No.	Topics	Links
Unit-I	Introduction to Visual Basic: VB IDE	https://www.nptelvideos.com/video.php?id=1771&c=21
	An overview of VB project types VB as event driven & object-based language	https://www.youtube.com/watch?v=hkcO_M9gcNw&t=20s
Unit-II	Working with Forms	https://www.youtube.com/watch?v=9EJXzWasTq4
	Dialog Boxes and Menu	https://www.youtube.com/watch?v=ww3nzKtuHqY
Unit-III	Database design and ER Model	https://www.nptelvideos.com/lecture.php?id=6473
	Integrity Constraint Over relations	https://www.nptelvideos.com/lecture.php?id=6474
Unit-IV	Lock based Concurrency	https://www.nptelvideos.com/lecture.php?id=6509
	Normalization: FIRST, SECOND, THIRD Normal forms, BCNF.	https://www.nptelvideos.com/lecture.php?id=6484

Course Code	**BC-MCA-125A**				
Category	Bridge Course LAB				
Course Title	**Software Lab-Bridge Course**				
Scheme and Credits	L	T	P	**Credits**	**SEMESTER-I**
	0	0	8	**4**	
Course Objectives	The objectives of this course are to • Understand the Basics of Computer Systems, Operating Systems, and Computer Networks. • Develop Proficiency in Using Office Automation Tools and Enhancing Productivity. • Learn C Programming Fundamentals and Master Problem-Solving Techniques • Understand and Apply Object-Oriented Programming Concepts in C++ • Develop Advanced C++ Programming Skills with Inheritance, Polymorphism, and Exception Handling • Design and Implement Efficient Algorithms and Data Structures				
Internal Assessment	50 Marks				
External Practical	50 Marks				
Total	100 Marks				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	**Skills Demonstrated**	**RBT Level**
CO1	Demonstrate the ability to apply programming logic and data handling techniques to solve real-world problems using C/C++.	Level 3: Apply
CO2	Break down complex problems into smaller parts and determine the most efficient approach to program design and data processing.	Level 4: Analyze
CO3	Critique and compare different methods of problem-solving, algorithm efficiency, and program functionality across various applications.	Level 5: Evaluate
CO4	Design and develop comprehensive software solutions, integrating various programming concepts and tools to meet user requirements.	Level 6: Create

List of Experiments

Sr. No.	Contents
1	Creating and Formatting Documents in MS Word
2	Data Analysis and Visualization in MS Excel
3	Creating and Delivering Presentations in MS PowerPoint
4	C Program to Find the Sum and Average of Three Numbers
5	C Program to Find the Sum of Individual Digits of a Given Integer
6	C Program to Generate Prime Numbers Between 1 to n
7	C Program to Check Whether a Given Number is Armstrong Number
8	C Program to Find Factorial of a Given Integer Using Non-Recursive Function
9	C Program to Find GCD of Two Integers Using Recursive Function
10	C Program to Find Both Largest and Smallest Number in a List
11	C++ Program to Print Multiplication Table
12	C++ Program to Find Factorial Using Recursive Function
13	C++ Program to Implement Inheritance, Polymorphism, Overloading
14	C++ Program for Sorting and Searching (e.g., Bubble Sort, Binary Search)
15	C++ Program to Implement Stacks Using Arrays

Suggested Readings:

- Gill Nasib Singh: Computing Fundamentals and Programming in C, Khanna Books Publishing Co., New Delhi, 2015.
- Kenneth. A.: C problem solving and programming, Prentice Hall, 1990.
- E. Balaguruswamy: Programming in C, Tata McGraw Hill, 2019.
- E. Balaguruswamy: C++, Tata McGraw Hill Publications, 2020.
- Herbert Schildt: C++ - The Complete Reference, Tata McGraw Hill Publications,2017

Virtual Lab Links:

Experiment No.	Experiment Name	Virtual Lab Links
1	Recursive Function	https://www.nptelvideos.com/lecture.php?id=6643
2	Inheritance	https://archive.nptel.ac.in/courses/106/105/106105151/
3	Polymorphism	https://www.youtube.com/watch?v=N0iYriJWq94&t=566s
4	Sorting and Searching	https://www.nptelvideos.com/lecture.php?id=6976

2nd Semester

Course Code	**DSC-MCA-102A**				
Category	Discipline-Specific Courses				
Course Title	**Advance Object Technology**				
Scheme and Credits	L	T	P	**Credits**	**Semester-II**
	4	0	0	**4**	
Course Objectives	The objectives of this course are to • Learn Java Script, XML, and applet basics for interactive web development. • Understand GUI development with Swing, AWT, and Java Beans. • Master Servlets and JSP for dynamic web content and MVC design. • Develop advanced JSP applications with error handling, data sharing, and frameworks.				
Assessment	40 Marks				
End Semester Examination	60 Marks				
Total Marks	100 Marks				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	**Skills Demonstrated**	**RBT Level**
CO1	Describe fundamental concepts of JavaScript, XML, applets, Swing, Servlets, JSP, and web development tools and frameworks.	Level 1: Remember
CO2	Explain the differences between AWT and Swing controls, JSP and Servlets, and understand the lifecycle of servlets and JSP pages.	Level 2: Understand
CO3	Implement interactive web pages and dynamic content using JavaScript, Servlets, JSP, and Swing, along with handling HTTP requests and responses.	Level 3: Apply
CO4	Analyze and debug the architecture of a web application using the MVC pattern, JSP, and Struts framework, ensuring optimal data sharing and memory management.	Level 4: Analyze

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 8 parts of 1.5 marks each from all units and remaining eight questions of 12 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit-I

Introduction to Java Scripts, Objects in Java Script, Dynamic HTML with Java Script. XML: Document type definition, XML Schemas, Document Object model, Presenting XML, Review of Applets, Class, Event Handling, AWT Programming.

Unit-II

Introduction to Swing, Differences between AWT Controls & Swing Controls, JApplet, Swing Button: JButton, JToggleButton, CheckBoxes, Radio Button, JComboBox, Text Boxes etc., Icons, Labels, JTabbed Pains, JScroll Pains, JList, JTrees, JTables Java Beans: Introduction to Java Beans, Advantages of Java Beans, BDK Introspection, developing a Home page using Applet & Swing.

Unit-III

Introduction to Servlets: Lifecycle of a Servlet, The Servlet API, The javax. Servlet Package, Reading Servlet parameters, Reading Initialization parameters; The javax.servlet HTTP package, Handling Http Request & Responses, Security Issues Introduction to JSP, Problem with Servlet. The Anatomy of a JSP Page, JSP Processing. JSP Application Design with MVC Setting Up and JSP Environment: Installing the Java Software Development Kit, Tomcat Server & Testing Tomcat.

Unit-IV

JSP Application Development: Generating Dynamic Content, Using Scripting Elements Implicit JSP Objects, Conditional Processing – Displaying Values Using an Expression to Set an Attribute, Declaring Variables and Methods Error Handling and Debugging Sharing Data Between JSP pages, Requests, and Users Passing Control and Date between Pages – Sharing Session and Application Data – Memory Usage Considerations Introduction to struts framework, RMI, CGI programming.

Suggested Readings:

- Dietel and Nieto: Internet and World Wide Web – How to program? PHI/Pearson Education Asia.
- Patrick Naughton and Herbert Schildt: The Complete Reference Java, Tata McGraw-Hill.
- Hans Bergstan: Java Server Pages.
- Bill Siggelkow, S P D O'Reilly: Jakarta Struts, Cookbook.
- Murach: Murach's beginning JAVA JDK 5, SPD.
- Wang-Thomson: An Introduction to Web Design and Programming.

Useful Video links: -

Unit No.	Topics	Links
Unit-I	Introduction to JavaScript:	https://archive.nptel.ac.in/courses/106/105/106105191/
	Java Applets and Event Handling.	https://youtu.be/0pzR2FGTEhk?si=1inrwjwWueDS-cyE
Unit-II	Introduction to Swing in Java	https://youtu.be/mQj34vUhpts?si=LeeznBdsLuXyJ7xJ
	AWT in Java	https://youtu.be/Vtc-eipVgL0?si=Q8-nxuOKIJVrAgop
Unit-III	HTTP package	https://youtu.be/-foyVzTOf8o?si=gYW1aNe7WdTuXSKc
Unit-IV	JSP Application Development	https://archive.nptel.ac.in/courses/106/106/106106156/

Course Code	**DSC-MCA-104A**				
Category	Discipline-Specific Courses				
Course Title	**Advance Database Systems and Data Warehouse**				
Scheme and Credits	L	T	P	**Credits**	**Semester-II**
	4	0	0	**4**	
Course Objectives	The objectives of this course are to • Understand advanced database models and their applications. • Learn parallel and distributed database technologies for optimized data processing. • Explore emerging database systems like mobile, multimedia, and GIS databases. • Master data warehousing and OLAP for effective data analysis and reporting.				
Assessment	40 Marks				
End Semester Examination	60 Marks				
Total Marks	100 Marks				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	Skills Demonstrated	RBT Level
CO1	Describe key concepts and terminology related to advanced database systems, including EER models, object-oriented databases, parallel databases, and data warehouses.	Level 1: Remember
CO2	Explain the principles of object-relational databases, distributed databases, active databases, and emerging database technologies like mobile, multimedia, and GIS databases.	Level 2: Understand
CO3	Implement advanced database models for object-oriented and distributed databases, including the design of EER models, object-relational database schemas, and client-server architectures.	Level 3: Apply
CO4	Analyze database architectures, query processing, transaction control, and concurrency management for distributed, parallel, and object-relational database systems.	Level 4: Analyze

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 8 parts of 1.5 marks each from all units and remaining eight questions of 12 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit-I

Introduction to Advance Database Systems: Overview of advance database systems, their importance and Applications;

EER Model -The ER model revisited, EER model: Super classes, Subclasses, Inheritance, Specialization and Generalization, Constraints and characteristics of specialization and Generalization, Category.

Object Model: Overview of Object-Oriented concepts, Object identity, Object structure, Type constructors, Encapsulation of operations, Methods, and Persistence, Type hierarchies and Inheritance, Complex objects, Schema design for OODBMS, OQL, Persistent Programming language, OODBMS architectures and storage issues, Transaction and concurrency control.

Object Relational Database and Information Retrieval: Database design for an ORDBMS – Nested relations and collections; Storage and access methods, Query processing and Optimization, Advance Querying: User defines data types, manipulating objects table, object views; Information Retrieval & ways to retrieve information.

Unit-II

Parallel Database: Architectures for parallel databases, Inter and Intra Query parallelism, Inter and Intra Query operations, Parallelizing individual operations, Sorting, Joins, and Pipelining.

Distributed Database: Architectures for Distributed Database, Data Fragmentation, Replication, and Allocation Techniques for Distributed Database Design, Query processing in Distributed Databases; Concurrency Control and Recovery in Distributed Databases.

Overview of Client Server Architectures: Centralized and Client-Server architectures, Server architectures.

Unit-III

Enhanced Data Models for Advanced Applications: Active database- syntax and semantics (DB2, Oracle), applications, design principles for active rules, Temporal database concepts, Spatial databases, Deductive databases.

Emerging Database Technologies: Mobile databases, Multimedia Databases, Geographic Information systems (GIS); XML and Internet Databases: Structured, Semi-structured and Unstructured Data, Introduction to web databases and XML, Structure of XML data.

Unit-IV

Data Warehouse and OLAP Technology: Need for data warehouse, Definition, Goals of data Warehouse, Challenges faced during Warehouse Construction, Advantages, Types of Warehouse: Data Mart, Virtual Warehouse and Enterprise Warehouse; Components of Warehouse: Fact data, Dimension data, Fact table and Dimension table, Designing fact tables; Pre-requisite Phases: Extract, Transform and load process; Warehouse Schema: star, snowflake and galaxy schemas; OLTP vs OLAP, Strengths of OLAP, Applications of OLAP.

Multidimensional data models: Data Cubes & Data Cuboids, Lattice; OLAP operations: Advantages, Types: Roll up, Drill down, Pivot, Slice & Dice operations, Applications; OLAP Server: Need, Types: ROLAP, MOLAP and HOLAP, Features; Data Warehouse Implementation, Introduction to Efficient computation of data cubes.

Suggested Readings:

- Elmasri and Navathe: Fundamentals of Database Systems, Pearson Education.
- Korth, Silberchatz, Sudarshan: Database System Concepts, McGraw-Hill.
- Raghu Ramakrishnan, Johannes Gehrke: Database Management Systems, McGraw-Hill
- Peter Rob and Coronel: Database Systems, Design, Implementation and Management, Thomson Learning.
- C.J.Date, Longman: Introduction to Database Systems, Pearson Education.

Useful Video links:

Unit No.	Topics	Links
Unit-I	Introduction to Advanced Database Systems:	https://www.nptelvideos.com/lecture.php?id=5903
	ER Model to Relational Mapping	https://www.nptelvideos.com/lecture.php?id=5910
	Object Oriented Databases	https://www.nptelvideos.com/lecture.php?id=5939
Unit-II	Query Processing and Optimization	https://www.nptelvideos.com/lecture.php?id=5917
	Concurrency Control for Distributed Transaction	https://www.nptelvideos.com/lecture.php?id=5929
	Introduction to Transaction Recovery	https://www.nptelvideos.com/lecture.php?id=5930
	XML - Introductory Concepts	https://www.nptelvideos.com/lecture.php?id=5941
	XML Databases	https://www.nptelvideos.com/lecture.php?id=5943
Unit-IV	Introduction to Data Warehousing and OLAP	https://www.nptelvideos.com/lecture.php?id=5933
	OLAP Operations: Roll Up, Drill Down, Pivot.	https://www.nptelvideos.com/lecture.php?id=5934

Course Code	**DSC-MCA-106A**				
Category	Discipline-Specific Courses				
Course Title	**Operating Systems and Shell Programming**				
Scheme and Credits	L	T	P	**Credits**	**Semester-II**
	4	0	0	**4**	
Course Objectives	The objectives of this course are to • Understand OS fundamentals: Process management, scheduling, and virtual machines. • Learn IPC and deadlock handling for efficient synchronization. • Master memory and I/O management for system optimization. • Develop Linux skills: Linux architecture, shell scripting, and utilities.				
Assessment	40 Marks				
End Semester Examination	60 Marks				
Total Marks	100 Marks				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	Skills Demonstrated	RBT Level
CO1	Describe fundamental concepts and terminology in operating systems, such as process management, memory management, interprocess communication, and file systems.	Level 1: Remember
CO2	Describe the functions, services, and structures of an operating system, as well as the principles of process scheduling, deadlock handling, and memory management.	Level 2: Understand
CO3	Apply scheduling algorithms, process management techniques, memory allocation strategies, and file management concepts to solve real-world operating system problems.	Level 3: Apply
CO4	Analyze various operating system problems, including deadlocks, interprocess communication issues, memory fragmentation, and I/O management challenges.	Level 4: Analyze

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 8 parts of 1.5 marks each from all units and remaining eight questions of 12 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit-I

Operating System Basics: Evolution, Objectives & Functions, Characteristics; Classification of Operating Systems, OS Services, System Calls, OS Structures, Concept of Virtual Machine.

Process Concepts: Definition, Process Relationship, Process states, Process State transitions, Process Control Block, Context switching – Threads – Concept of multithreads, Benefits of threads – Types of threads.

Process Scheduling: Definition, Scheduling objectives, Types of Schedulers, Scheduling criteria. Scheduling Algorithms: Preemptive and Non-preemptive, FCFS–SJF–RR, Multiprocessor scheduling: Types, Performance evaluation of the scheduling.

Unit-II

Interprocess Communication: Race Conditions, Critical Section, Mutual Exclusion, Hardware Solution, Strict Alternation, Peterson's Solution, Producer Consumer Problem, Semaphores, Event Counters, Monitors, Message Passing, Classical IPC Problems: Reader's & Writer Problem, Dinning Philosopher Problem.

Deadlocks: System Model, Deadlock Principles, Deadlock Characterization, Methods for Handling Deadlocks Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock.

Unit-III

Memory Management: Basic Memory Management, Logical and Physical address map, Memory allocation, Fragmentation and Compaction, Paging and its disadvantages, Virtual Memory, Locality of reference, Page Fault, Working Set, Demand paging concept, Page Replacement policies.

Input/Output Management: I/O devices, Device controllers, Direct memory access Principles of I/O Software: Goals of Interrupt handlers, Device drivers, Device independent I/O software, Secondary-Storage Structure: Disk structure, Disk scheduling algorithms.

File Management: File concept, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods.

Unit-IV

Linux Basics: Genesis of Linux, Architecture of Linux, Features of Linux, Introduction to vi editor, Linux commands. Linux Shells: Role, Types- Bourne Shell (sh), C Shell (csh), Korn Shell (ksh), Bourne Again Shell (bash).

Linux Utilities: File handling utilities, Security by file permissions, Process utilities, Disk utilities, Networking commands, Filters, Text Processing utilities and backup utilities.

Shell programming (With bash): Introduction, shell responsibilities, pipes and Redirection, running a shell script, the shell as a programming language, Shell meta characters, File name substitution, Shell variables, Command substitution, Shell commands, the environment, Quoting, Test command, control structures, arithmetic in shell, shell script examples, interrupt processing, functions, debugging shell scripts.

Suggested Readings:

- Dhamdhere: Operating Systems, Tata McGraw Hill.
- Robert Love: Linux System Programming, O'Reilly, SPD.
- Jason Cannon: Linux For Beginners,
- William Shotts: The Linux Command Line: A Complete Introduction.
- Daniel J. Barrett: Linux Pocket Guide: Essential Commands
- Any other book(s) covering the contents of the paper in more depth.
- Silberschatz& Galvin: Operating System Concept, Wiley.
- Milan Milenkovic: Operating Systems, Tata McGraw – Hill.

Useful Video links: -

Unit No.	Topics	Links
Unit-I	Operating System Overview	https://www.youtube.com/watch?v=a2B69vCtjOU
	Process states	https://www.youtube.com/watch?v=jENkpxUOoXM&list=PLyqSpQzTE6M9SYI5RqwFYtFYab94gJpWk&index=4
Unit-II	Deadlock Prevention and Avoidance	https://www.youtube.com/watch?v=UczJ7misUEk&list=PLyqSpQzTE6M9SYI5RqwFYtFYab94gJpWk&index=32
Unit-III	Memory Management Techniques	https://www.youtube.com/watch?v=_pKbqFyG03s&list=PLyqSpQzTE6M9SYI5RqwFYtFYab94gJpWk&index=6
Unit-IV	Introduction to Linux and Shells	https://archive.nptel.ac.in/courses/117/106/117106113/
	Networking commands	https://www.youtube.com/watch?v=Z0RVnURLRZM&t=334s

Course Code	**DSC-MCA-112A**				
Category	Discipline Specific Elective Courses				
Course Title	**Software Engineering**				
Scheme and Credits	L	T	P	**Credits**	**Semester-II**
	4	0	0	**4**	
Course Objectives	The objectives of this course are to • Understand software development models and apply them to various projects. • Design, implement, and test software solutions using appropriate strategies and techniques. • Manage software projects effectively, focusing on risk management, estimation, and maintenance.				
Assessment	40 Marks				
End Semester Examination	60 Marks				
Total Marks	100 Marks				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	Skills Demonstrated	RBT Level
CO1	Describe fundamental concepts of software engineering, including software life cycle models, requirements gathering, software design principles, and types of software maintenance.	Level 1: Remember
CO2	Explain the various software life cycle models, software requirements, design strategies, and the differences between verification and validation in software testing.	Level 2: Understand
CO3	Apply software engineering practices like requirement analysis, system design, testing, and project management strategies.	Level 3: Apply
CO4	Analyze design, testing, and project plans to identify improvements using metrics like LOC and complexity.	Level 4: Analyze

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 8 parts of 1.5 marks each from all units and remaining eight questions of 12 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit-I

Introduction: Software and its Characteristics, Evolving Role of Software, Software Product, Software Processes, Software Crisis, 'Software Engineering' Evolution, Principles of Software Engineering, Programming-in-the-small vs. Programming-in-the-large, Software Components, Software Engineering Processes.

Software Life Cycle (SLC) Models: Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models, Object Oriented Models and other latest Models.

Software Requirements: Functional and Non-Functional, User requirements, System requirements. Software Requirements Document – Requirement

Engineering Process: Feasibility Studies, Requirements elicitation and analysis, requirements validation, requirements management.

Unit-II

Software Design: Basic Concept of Software Design, Architectural Design,

Low Level Design: Modularization, Design Structure Charts, Flow Charts, Coupling and Cohesion Measures; Design Strategies: Function Oriented Design, Object Oriented Design, Top-Down and Bottom-Up Design, User Interface Design, Programming practices and Coding standards.

Software Testing: Introduction, Verification vs. Validation, Software Reliability, Levels of Testing, Structural Testing (White Box Testing), Functional Testing (Black Box Testing).

Unit-III

Software Quality: Attributes, Software Quality Assurance – plans & activities; Software Documentation.

Software Project Management: Project Management activities, Project Estimation, Project planning, Project scheduling.

Software Risk Management: Reactive versus Proactive Risk Strategies, Risk management activities; Software Risks (Risk Identification, Risk Projection, Risk Refinement, Risk Mitigation), Risks Monitoring and Management.

Software Measurement and Metrics: Process Metrics, Project metrics, Estimation – LOC, Halstead's Software Science, Function Point (FP), Cyclomatic Complexity Measures; Software Project Estimation Models- Empirical, Putnam, COCOMO I & II.

Unit-IV

Software Maintenance: Need for Maintenance, Categories of Maintenance: Preventive, Corrective and Perfective Maintenance, Cost of Maintenance; Software Re- Engineering, Reverse Engineering, Software Documentation.

Software Configuration Management: SCM Activities, Change Control Process, Software Version Control; Software Reuse, Software Evolution.

CASE Computer Aided Software Engineering (CASE), CASE Tools.

Suggested Readings:

- Rogers Pressman: Software Engineering, TMH.
- Gill, Nasib Singh: Software Engineering, Khanna Book Publishing Co.(P) Ltd, New Delhi
- Jalote, Pankaj: An Integrated Approach to Software Engineering, Narosa Publications.
- Chhillar Rajender Singh: Software Engineering: Testing, Faults, Metrics, Excel Books, New Delhi.
- Ghezzi, Carlo: Fundaments of Software Engineering, PHI.

Useful Video links

Unit No.	Topics	Links
Unit-I	Introduction to Software Engineering	https://www.nptelvideos.com/lecture.php?id=7029
	Software Engineering-Challenges	https://www.nptelvideos.com/lecture.php?id=7028
	Requirements Engineering / Specification	https://www.nptelvideos.com/lecture.php?id=7032
Unit-II	Software Design Techniques	https://www.nptelvideos.com/lecture.php?id=7041
	Software Testing Overview	https://www.nptelvideos.com/lecture.php?id=7045
	White Box and Black Box Testing	https://www.nptelvideos.com/lecture.php?id=7046
Unit-III	Software Quality Management Systems	https://www.nptelvideos.com/lecture.php?id=7063
	Software Maintenance and Reengineering	https://www.youtube.com/watch?v=DW78Vr6Y2fg
Unit-IV	Software Management	https://www.nptelvideos.com/lecture.php?id=7065

Course Code	**DSEC-MCA-116A**				
Category	Discipline Specific Elective Courses				
Course Title	**Computer Networks and Distributed Systems**				
Scheme and Credits	L	T	P	**Credits**	**Semester-II**
	4	0	0	**4**	
Course Objectives	The objectives of this course are to • Understand the fundamentals of data communication, including network models and protocols. • Comprehend the principles of distributed systems, focusing on synchronization, fault tolerance, and security. • Analyze network and transport layer protocols, data link functions, and implement practical communication mechanisms.				
Assessment	40 Marks				
End Semester Examination	60 Marks				
Total Marks	100 Marks				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	**Skills Demonstrated**	**RBT Level**
CO1	Describe the fundamental concepts of data communication, network classification, communication models, and protocols used in distributed systems.	Level 1: Remember
CO2	Explain data encoding, modulation techniques, multiplexing, and the layered architecture of network reference models system, communication models, and fault tolerance mechanisms.	Level 2: Understand
CO3	Implement various data link layer protocols, such as HDLC and sliding window protocol, and apply network protocols and replication management.	Level 3: Apply
CO4	Analyze data link protocols, error correction, medium access, synchronization, and fault tolerance in distributed systems.	Level 4: Analyze

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 8 parts of 1.5 marks each from all units and remaining eight questions of 12 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit-I

Data Communication: Introduction to data communication; analog and digital signals; asynchronous and synchronous transmission; data encoding and modulation techniques, broadband and base band transmission, multiplexing, transmission medium.

Network Classification: Wired Network, Wireless Network, Internetworking Devices.

Network Reference Models: Layered architectures, protocol hierarchies, interface and services: ISO- OSI reference model, TCP/IP reference model; internet protocol stacks.

Unit-II

Data Link Layer Functions and Protocols: Framing, error-control, flow-control; sliding window protocol; HDLC, Error detection and correction, Data link layer of internet.

Medium Access Sub-layer: CSMA/CD protocol, IEEE standards for LAN and MAN, X.25, frame relay, Narrowband and Broadband ISDN, Asynchronous Transfer Modes.

Network Functions & Protocols: Switching mechanism and its various types, routing and congestion control, Internetworking-TCP/IP, IP Packet, IP address, IPv6

Transport Layer: Design issues, Connection management (UDP, TCP and their Frame Format); Application Layer: File Transfer, Access and Management, E-Mail, Virtual Terminal, Public Network

Unit-III

Introduction to Distributed Systems: Introduction, Design Goals, Types of Distributed systems, System Architecture and Fundamental models, Middleware, Threads, Virtualization, Client-server model, multiple servers, proxy servers and caches, peer processes, code migration.

Communication Fundamentals: Basic concepts, Remote Procedure Call, Message Oriented Communication, Stream Oriented Communication, Multicast Communication.

Synchronization: Clock synchronization, Logical clocks, Mutual exclusion algorithms: centralized, decentralized, distributed and token ring algorithms, election algorithms.

Unit-IV

Replication Management: Need for replication, Consistency models, Consistency protocols, Replica management.

Fault Tolerance: Basic concepts and failure models, Process resilience, Reliable client-server and group communication, Distributed commit recovery mechanisms.

Security in Distributed Systems: Secure channels, Access control, Security management, Cryptographic algorithms; Digital signatures; certificates, firewalls.

Naming: Flat naming, Structured naming, Name space and Resolution, Attribute- based naming, Directory services, LDAP, Decentralized implementations.

Case Studies: Needham-Schroeder, Kerberos, SSL. operations: Advantages, Types: Roll up, Drill down, Pivot, Slice & Dice operations, Applications; OLAP Server: Need, Types: ROLAP, MOLAP and HOLAP, Features; Data Warehouse Implementation, Introduction to Efficient computation of data cubes.

Suggested Readings:

- W. Tomasi: Introduction to Data Communications and Networking, Pearson Education.
- P.C. Gupta: Data Communications and Computer Networks, Prentice-Hall of India.
- Behrouz Forouzan and S.C. Fegan: Data Communications and Networking, McGraw Hill.
- L. L. Peterson and B. S. Davie: Computer Networks: A Systems Approach, Morgan
- Kaufmann.
- William Stallings: Data and Computer Communications, Pearson Education.
- George Coulouris, Jean Dollimore, Tim Kindberg: Distributed Systems-Concepts and Design, Pearson Education

Useful Video links:

Unit No.	Topics	Links
Unit-I	Introduction to Data Communication	https://www.nptelvideos.com/lecture.php?id=5826
	Transmission Impairments and Channel Capacity	https://www.nptelvideos.com/lecture.php?id=5829
	Introduction to Data Transmission & Networking	https://www.nptelvideos.com/lecture.php?id=5860
Unit-II	IEEE 802 LANs	https://www.nptelvideos.com/lecture.php?id=5854
	TCP/IP - II	https://www.nptelvideos.com/lecture.php?id=5861
	X.25 and Frame Relay	https://www.nptelvideos.com/lecture.php?id=5849
Unit-III	Introduction to Distributed Systems	https://archive.nptel.ac.in/courses/106/106/106106168/
	Remote Procedure Call and Communication	https://www.youtube.com/watch?v=wqzKqYp
Unit-IV	Firewalls and Intrusion Detection Systems	https://www.nptelvideos.com/lecture.php?id=5824

Course Code	**EL-MCA-108A**				
Category	Experiential Learning				
Course Title	**Software Lab-3**				
Scheme and Credits	L	T	P	**Credits**	**Semester-II**
	0	0	6	**3**	
Course Objectives	The objectives of this course are to • Explore emerging database systems like mobile, multimedia, and GIS databases. • Master data warehousing and OLAP for effective data analysis and reporting.				
Assessment	50 Marks				
End Semester Practical Examination	50 Marks				
Total Marks	100 Marks				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	**Skills Demonstrated**	**RBT Level**
CO1	Apply SQL commands to insert, update, delete records, and perform calculations on columns using arithmetic operations and SQL functions.	Level 3: Apply
CO2	Analyze the data using logical operators, BETWEEN, and aggregate functions such as COUNT and GROUP BY to group and filter records.	Level 4: Analyze
CO3	Evaluate the structure of SQL queries for data integrity and correctness, including proper use of joins, grouping, and ordering of records.	Level 5: Evaluate
CO4	Design and create views, implement constraints, and demonstrate effective transaction control to ensure database consistency and integrity.	Level 6: Create

List of Experiments

Sr. No.	Contents
1	Write SQL command to create a table with columns of your choice. Write an SQL command to add a new column to the table. Write an SQL command to delete the table from the database.
2	Write SQL command to insert a new record into the table. Write an SQL command to update a column value in the table. Write an SQL command to delete a record from the table.
3	Write SQL commands showcase the use `ROUND`, `COUNT` , `UPPER` , `SYSDATE` , and `TO_CHAR` functions
4	Write SQL command to perform an arithmetic calculation on numeric columns.
5	Write SQL command to find records that satisfy either of two conditions using logical operators.
6	Write SQL command using the `BETWEEN` operator to find records within a specific range.
7	Write SQL command to perform a natural join between two tables.
8	Write SQL command to group records by a column and count the number of records in each group, displaying only groups that meet a specific condition.
9	Write SQL command to order records by a specific column in ascending or descending order.
10	Write SQL command to create a view that displays specific columns from a table.
11	Write SQL commands to implement `PRIMARY KEY`, `FOREIGN KEY`, `UNIQUE`, `CHECK`, and `NOT NULL` constraints on a table.
12	Write SQL commands to demonstrate transaction control using `ROLLBACK`, `COMMIT`, and `SAVEPOINT`.

Useful Video Link:

Experiment No	Experiment Name	Links
1	Write SQL command to create a table with columns of your choice. Write an SQL command to add a new column to the table. Write an SQL command to delete the table from the database.	https://archive.nptel.ac.in/courses/106/106/106106220/
2	Write SQL command to insert a new record into the table. Write an SQL command to update a column value in the table. Write an SQL command to delete a record from the table.	https://archive.nptel.ac.in/courses/106/106/106106220/
3	Write SQL commands showcase the use `ROUND`, `COUNT`, `UPPER`, `SYSDATE`, and `TO_CHAR` functions	https://archive.nptel.ac.in/courses/106/106/106106220/
4	Write SQL command to perform an arithmetic calculation on numeric columns.	https://archive.nptel.ac.in/courses/106/106/106106220/
5	Write SQL command to find records that satisfy either of two conditions using logical operators.	https://archive.nptel.ac.in/courses/106/106/106106220/
6	Write SQL command to create a table with columns of your choice. Write an SQL command to add a new column to the table. Write an SQL command to delete the table from the database.	https://archive.nptel.ac.in/courses/106/106/106106220/

Course Code	**EL-MCA-110A**				
Category	Experiential Learning				
Course Title	**Software Lab-4**				
Scheme and Credits	L	T	P	**Credits**	**Semester-II**
	0	0	6	**3**	
Course Objectives	The objectives of this course are to • Understand basic concepts of Operating Systems and their structure. • Learn about concept of processes and process scheduling • Learn in detail about Deadlock, memory management and I/O management.				
Assessment	50 Marks				
End Semester Practical Examination	50 Marks				
Total Marks	100 Marks				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	**Skills Demonstrated**	**RBT Level**
CO1	Implement object-oriented concepts such as inheritance, abstraction, interfaces, event handling, and graphics algorithms in Java and C/C++.	Level 3: Apply
CO2	Analyze and debug Java programs, algorithms for geometric transformations, and multi-threading applications for optimization and error resolution.	Level 4: Analyze
CO3	Evaluate the efficiency and functionality of matrix multiplication, event handling techniques, graphical algorithms, and multi-threading in Java and C/C++.	Level 5: Evaluate
CO4	Design and develop Java applications and C/C++ programs for graphical transformations, event-driven programming, and multi-threading with real-time functionality.	Level 6: Create

List of Experiments:

Sr. No.	Contents
1	To understand and run basic UNIX/LINUX commands
2	To study of Basic Unix/Linux Commands and various Unix/Linux editors such as vi, ed, ex and EMACS.
3	To study basic file manipulation commands of Unix/Linux commands
4	To write C Programs using the following system calls of Unix/Linux operating system fork, exec, getpid, exit, wait.
5	To write C Programs using the following system calls of Unix/Linux operating system close, stat, opendir, readdir.
6	To write C programs to simulate Unix/Linux commands like cp, ls, grep.
7	Write a Shell program to check the given number is even or odd.
8	Write a Shell program to check the given year is leap year or not.
9	Write a Shell program to find the factorial of a number.
10	Write a C program for implementation of Priority scheduling algorithms.
11	To write a C program for implementation of Round Robin scheduling algorithms.
12	To write a C program for implementation of FCFS and SJF scheduling algorithms.

Useful Video Links:

Experiment No.	Experiment Name	Links
1	To understand and run basic UNIX/LINUX commands	https://www.youtube.com/watch?v=rRGCGZ6OHw8&t=612s
2	To study basic file manipulation commands of Unix/Linux commands	https://www.youtube.com/watch?v=Z0Vkrn9faoM&t=510s
3	Networking commands	https://www.youtube.com/watch?v=Z0RVnURLRZM&t=334s

Course Code	**DSEC-MCA-113A**				
Category	Discipline Specific Elective Courses				
Course Title	**Cloud Computing**				
Scheme and Credits	L	T	P	**Credits**	**Semester-I**
	4	0	0	**4**	
Course Objectives	The objectives of this course are to • Understand Cloud Fundamentals: Grasp the basic concepts, models, architecture, and benefits of cloud computing. • Explore Virtualization and Platforms: Study the role of virtualization and popular cloud platforms like AWS, Google Cloud, and Azure. • Learn Cloud Management: Gain insights into cloud service management, data scalability, and service-oriented architectures.				
Assessment	40 Marks				
End Semester Examination	60 Marks				
Total Marks	100 Marks				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	Skills Demonstrated	RBT Level
CO1	Describe cloud computing concepts, deployment models, service models, and platforms (IaaS, PaaS, SaaS) with features.	Level 1: Remember
CO2	Explain cloud computing concepts, architecture, benefits, virtualization, load balancing, and platforms like AWS, Google Cloud, Azure.	Level 2: Understand
CO3	Apply cloud services (IaaS, PaaS, SaaS) and platforms like AWS, Google App Engine, Azure for deployment.	Level 3: Apply
CO4	Analyze cloud architecture, compare service models, and evaluate benefits/limitations of virtualization, SDN, and NFV.	Level 4: Analyze

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 8 parts of 1.5 marks each from all units and remaining eight questions of 12 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit-I

Cloud Computing Fundamentals: Definition of Cloud Computing: Defining a cloud, Evolution of Cloud Computing cloud types-NIST model, cloud cube model, Deployment models, Service models, Cloud Reference model, Characteristics of Cloud, Cloud Computing Benefits and Limitations, Cloud Architecture: Introduction on Infrastructure, platforms, virtual appliances, communication protocols; Cloud computing vs. Cluster computing vs. Grid computing; Applications: Technologies and Process required when deploying Web services; Deploying a web service from inside and Outside of a Cloud. Services and Applications by Types: IaaS, PaaS, SaaS, IDaaS, and CaaS.

Unit-II

Virtualization: Objectives, Benefits of Virtualization, Emulation, Virtualization for Enterprise, VMware, Server Virtualization, Data Storage Virtualization, Load balancing and Virtualization, Improving Performance through

Load Balancing, Hypervisors, Machine Imaging, Porting of applications in the cloud. Concept of Software-Defined Networking (**SDN**), Network-Function Virtualization (**NFV**) and Virtual Network Functions (**VNF**).

Use of Platforms in Cloud Computing: Concepts of Platform as a Service, Use of PaaS application frameworks; Use of Google, Amazon and Microsoft Web Services. Cloud vendors and Service Management: Amazon cloud, AWS Overview, Installation of AWS, Google app engine, azure cloud, salesforce.

Unit-III

Cloud Management: Features of Network management system, Monitoring of an entire cloud computing deployment stack, lifecycle management of cloud services(six stages of lifecycle)

Service Management in Cloud Computing: Service Oriented Architecture: concepts of message-based transactions, Protocol stack for an SOA architecture, Event driven SOA, Enterprise Service bus, Service Catalogs, Service Level Agreements (SLAs), Managing Data: Looking at Data, Scalability & Cloud Services, Database & Data Stores in Cloud , Large Scale Data Processing.

Unit-IV

Cloud Security Concepts: Cloud security challenges, Cloud security approaches: encryption, tokenization/ obfuscation, cloud security alliance standards, cloud security models and related patterns, Cloud security in mainstream vendor solutions, Mainstream Cloud security offerings: security assessment, secure Cloud architecture design, Authentication in cloud computing, Client access in cloud, Cloud contracting Model, Security Mapping, Identity Management.

Case Study on Open Source & Commercial Clouds: Eucalyptus, Microsoft Azure, Amazon EC2.

Suggested Readings:

- Cloud Computing: A Practical Approach by Anthony T. Velte Toby J. Velte, Robert Elsenpeter, The McGraw-Hill.
- Cloud Computing: SaaS, PaaS, IaaS, Virtualization and more. by Dr. Kris Jamsa.
- Tim Mather, Subra Kumara swamy, Shahed Latif: Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, O'ReillyMedia Inc.
- Cloud Computing Bible, Barrie Sosinsky, Wiley-India.
- Jason Venner,Pro: Hadoop,Apress.
- Cloud Computing: Principles and Paradigms, Editors: Raj kumar Buyya, James Broberg, Andrzej M. Goscinski, Wiley.

Useful Video links:

Unit No.	Topics	Links
Unit-I	Cloud Computing Fundamentals	https://youtu.be/NzZXz3fJf6o?si=Jl6S2DP0_NmiJn2P
	Cloud Computing Architecture	https://youtu.be/fZ3D6HQrWzs?si=bugPgty_2y9UC-J2
Unit-II	Introduction to Virtualization and Cloud Computing	https://archive.nptel.ac.in/courses/106/105/106105167/
	Virtualization Explained	https://youtu.be/GtJGB1WxRW8?si=JIiaURE5tKb23X_g
	Cloud Platforms Overview (AWS, Azure, Google Cloud	https://youtu.be/XTW3ujxnndo?si=zTX1WCHmbsH1P9I1
Unit-III	Cloud Management Essentials	https://archive.nptel.ac.in/courses/106/104/106104182/
	Database Management and Cloud Scalability	https://youtu.be/CYY0lRKQwb8?si=h0P_Ac-3R8-Ci7UL
Unit-IV	Cloud Security Basics	https://archive.nptel.ac.in/courses/106/105/106105223/
	Quantum Computing and Security in the Cloud	https://archive.nptel.ac.in/courses/106/104/106104242/

Course Code	**DSEC-MCA-115A**				
Category	Discipline Specific Elective Courses				
Course Title	**Theory of Computation**				
Scheme and Credits	L	T	P	**Credits**	**Semester-II**
	4	0	0	**4**	
Course Objectives	The objectives of this course are to • Grasp mathematical foundations for formalizing computation. • Design automata and grammars for language recognition. • Analyze computational models to understand decidability. • Classify problems by computational complexity (P, NP).				
Assessment	40 Marks				
End Semester Examination	60 Marks				
Total Marks	100 Marks				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	Skills Demonstrated	RBT Level
CO1	Describe key concepts in formal languages, automata theory, Turing machines, and computational complexity properties.	Level 1: Remember
CO2	Explain the relationship between formal languages, grammars, automata, Chomsky hierarchy, undecidability, and Turing machines.	Level 2: Understand
CO3	Apply automata theory to design finite automata, pushdown automata, Turing machines, and solve with pumping lemma.	Level 3: Apply
CO4	Analyze languages to simplify grammars, convert to normal forms, and evaluate equivalence and decidability.	Level 4: Analyze

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 8 parts of 1.5 marks each from all units and remaining eight questions of 12 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit-I

Review of Mathematical Terms and Theory: Basic Mathematical Notations and Set Theory, Logic Functions And Relations, Language Definitions, Mathematical Inductions and Recursive Definitions.

Finite Automata: Introduction, Alphabets, Strings and Languages, Kleen-closure; Deterministic Finite Automata (DFA) and Nondeterministic Finite Automata (NFA) -Formal definition, simpler notations (state transition diagram, transition table), Regular and Non-Regular Languages, Equivalence of NDFA & DFA, NDFA to DFA conversion, DFA minimization using Myhill-Nerode Theorem, Applications of Finite Automata, Finite automata with output (Moore and Mealy machines) and inter-conversion.

Unit-II

Context Free Grammar: Introduction to CFG, CFG and Known Languages, Unions Concatenations and *'S Notations and CFL, Derivations of Trees and Ambiguity, Unambiguous CFG and Algebraic Expressions, Normal Forms and Simplified Forms.

Formal Grammar: Definition, Chomsky hierarchy of grammars, Construction of Context free, derivation, parse tree, ambiguity in grammars, Removal of null and unit production, Normal forms- CNF & GNF.

Pushdown Automata: Introduction to PDA, Types of PDA, Designing of PDA, CFG Corresponding to PDA, Introduction to CFL, Intersections and Complements of CFL, Decisions Problems and CFL, Equivalence of Pushdown Automata and CFL, Pumping Lemma for CFL, Applications.

Unit-III

Turing Machines: Model of Computation and Church Turning Thesis, Definition of Turing Machine, Tm and Language Acceptors, Variations of Tm, Non- Deterministic Tm, Universal Tm, Tm & computers.

Recursive Language: Introduction, Enumerable and Language, Recursive and Non Recursive Enumerable, their properties.

PCP: Introduction to undecidability, undecidable problems about TMs, Post correspondence problem (PCP), Modified PCP.

Unit-IV

Computation Functions, Measuring, Classifications and Complexity: Primitive Recursive Functions, Halting Problem, Recursive Predicates and Some Bounded Operations, Unbounded Minimizations and μ-Recursive Functions, Godel Numbering, Computable Functions and μ-Recursive, Numerical Functions.

Tractable and Intractable Problems: Growth Rate and Functions, Time and Speed Complexity, Complexity Classes, Tractable and Possibly Intractable Problems, P And NP Completeness, Reduction Of Time, Cook's Theorem, NP-Complete Problems.

Suggested Readings:

- John C. Martin: Introduction to Language and theory of Computation, Mcgraw Hill.
- John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman: Introduction to Automata Theory Languages and Computation, Pearson Education
- K. L. P Mishra, N. Chandrashekaran: Theory of Computer Science-Automata Languages and Computation, Prentice Hall of India, India.
- K.Krithivasan and R.Rama: Introduction to Formal Languages, Automata Theory and Computation; Pearson Education.
- Harry R. Lewis and Christos H. Papadimitriou: Elements of the Theory of Computation, Second Edition, Prentice-Hall of India Pvt. Ltd
- Any other book(s) covering the contents of the paper in more depth. 6. Wang-Thomson: An Introduction to Web Design and Programming.

Useful Video links

Unit No.	Topics	Links
Unit-I	Introduction to Set Theory	https://www.nptelvideos.com/lecture.php?id=7192
	Finite Automata and Regular Languages	https://www.nptelvideos.com/lecture.php?id=7193
Unit-II	Context-Free Grammars (CFGs)	https://www.nptelvideos.com/lecture.php?id=7211
	Pushdown Automata	https://www.nptelvideos.com/lecture.php?id=7222
Unit-III	Turing Machines	https://www.nptelvideos.com/lecture.php?id=7229
	Non- Deterministic Tm	https://youtu.be/RG5bohlmaTk?si=9q7rmnS6yBbm4c0O
	Post correspondence problem (PCP)	https://youtu.be/TzI3DXEXBwE?si=rsONlMBx6-MqME4W
Unit-IV	Time and Speed Complexity	https://youtu.be/Jsh5iwNfmIQ?si=u93h6mq8oOhxSOEj
	Cook's Theorem	https://youtu.be/96JURPTS8dA?si=pVtHYjJc2L3dv_d3
	NP-Complete Problems	https://youtu.be/Wp_nB-MF_Mk?si=3EtRW5j_BxV7RKDa

Course Code	**DSEC-MCA-114A**				
Category	Discipline Specific Elective Courses				
Course Title	**Web Technologies**				
Scheme and Credits	L	T	P	**Credits**	**Semester-I**
	4	0	0	**4**	
Course Objectives	The objectives of this course are to • Understand Web Architecture: Learn web protocols, HTML, XHTML, and CSS for web page design. • JavaScript Programming: Gain skills in client-side JavaScript for interactivity and DOM manipulation. • · Server-Side Programming: Develop dynamic web content using Java Servlets and manage sessions.				
Assessment	40 Marks				
End Semester Examination	60 Marks				
Total Marks	100 Marks				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	**Skills Demonstrated**	**RBT Level**
CO1	Describe web development concepts: HTML, CSS, JavaScript, Servlets, JSP, XML, AJAX, DOM, HTTP, W3C.	Level 1: Remember
CO2	Explain web clients, servers, protocols, and the role of HTML, CSS, JavaScript, Servlets, JSP.	Level 2: Understand
CO3	Apply static/dynamic web pages with HTML, CSS, JavaScript, and create applications using Servlets, JSP, AJAX.	Level 3: Apply
CO4	Analyze client-side vs server-side technologies and evaluate integrating HTML, CSS, JavaScript, XML, AJAX, Web Services.	Level 4: Analyze

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 8 parts of 1.5 marks each from all units and remaining eight questions of 12 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit-I

Web Essentials: Clients, Servers, and Communication. Basic Internet Protocols, W3C, HTTP, Web Clients & Web Servers.

Markup languages-XHTML: Introduction to HTML, basics of XTHML, HTML elements, HTML tags, lists, tables, frames, forms, defining XHTML's abstract syntax, defining HTML documents.

CSS style sheets: Introduction, CSS core syntax, text properties, CSS box model, normal flow box layout, other properties like list, tables, DHTML, XML, XML documents & vocabulary, XML versions & declarations, Introduction to WML.

Unit-II

Client-Side Programming: The JavaScript Language-History and Versions Introduction JavaScript in Perspective-Syntax-Variables and Data Types-Statements-Operators-Literals-Functions-Objects-Arrays-Built-in Objects-JavaScript Debuggers.

Host Objects: Browsers and the Document Object Model (DOM), Levels-Intrinsic Event Handling-Modifying Element Style-The Document Tree-DOM Event Handling-Accommodating Noncompliant Browsers Properties of window.

Server-Side Programming: Concept of server-side programming, Java Servlets revistited- Architecture - Overview-A Servlet-Generating Dynamic Content-Life Cycle- Parameter Data-Sessions-Cookies-URL Rewriting-Other Capabilities-Data Storage Servlets and Concurrency- Databases and Java Servlets.

Unit-III

Separating Programming and Presentation: JSP Technology revisited - JavaBeans Classes and JSP-Tag Libraries and Files-Support for the Model-View-Controller Paradigm- Databases and JSP. Representing Web Data: XML-Documents and Vocabularies-Versions and Declaration-Namespaces- DOM based XML processing Event-oriented Parsing: SAX-Transforming XML Documents-Selecting XML Data: XPATH-Template based Transformations: XSLT-Displaying XML Documents in Browsers.

Unit-IV

AJAX: Ajax Client Server Architecture-XML Http Request Object-Call Back Methods.

Web Services: JAX-RPC-Concepts-Writing a Java Web Service-Writing a Java Web Service Client-Describing Web Services: WSDL- Representing Data Types: XML Schema-Communicating Object Data: SOAP Related Technologies-Software Installation-Storing Java Objects as Files.

Suggested Readings:

- Jackson: Web Technologies: A Computer Science Perspective, Pearson Education India.
- Roger S Pressman, David Lowe: Web Engineering: A Practitioner's Approach, TMH.
- Achyut Godbole, Atul Kahate: Web Technologies, McGraw-Hill Education.
- Uttam K Roy: Web Technologies, Oxford University Press.
- Chris Bates: Web Programming, Wiley.
- Gertel Keppel, Birgit Proll, Siegfried Reich, Werner R.: Web Engineering, John Wiley& Sons Inc.

Useful Video links:

Unit No.	Topics	Links
Unit-I	Basic Internet Protocols	https://youtu.be/nhGRUmtnGB4?si=Yaic70rQ5Lu-KSvk
	Introduction to HTML	https://youtu.be/QEtWL4lWlL4?si=Ohu2kd6ucbnyg4xc
Unit-II	JavaScript Language-History and Versions	https://youtu.be/uUhOEj4z8Fo?si=cFFEWSGScuBHvFB9
	Java Servlets revisited- Architecture - Overview	https://youtu.be/XlryaovT_3k?si=rxJliGNfb67hhiv2

Course Code	**DSEC-MCA-118A**				
Category	Discipline Specific Elective Courses				
Course Title	**Advance Computer Architecture & Quantum Computing**				
Scheme and Credits	L	T	P	**Credits**	**Semester-II**
	4	0	0	**4**	
Course Objectives	The objectives of this course are to • Understand Computer Architecture: Learn the evolution, classification, and parallelism concepts in computer architectures. • Analyze System Interconnects and Memory: Explore interconnect networks, memory hierarchy, and caching to optimize computing performance. • Examine Multiprocessor Architectures: Study multiprocessor systems, shared memory models, and message-passing techniques.				
Assessment	40 Marks				
End Semester Examination	60 Marks				
Total Marks	100 Marks				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	Skills Demonstrated	RBT Level
CO1	Describe computer architecture concepts, evolution, system attributes, parallel computers, network architectures, and memory hierarchies.	Level 1: Remember
CO2	Explain parallelism conditions, dependencies, hardware/software parallelism, and compare control flow, data flow mechanisms.	Level 2: Understand
CO3	Apply system interconnects, static/dynamic networks, and optimize performance with memory hierarchy, cache, virtual memory.	Level 3: Apply
CO4	Analyze multiprocessor and multicomputer architectures, shared/distributed memory, and interconnects like buses, crossbar, multiport memory.	Level 4: Analyze

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 8 parts of 1.5 marks each from all units and remaining eight questions of 12 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit-I

Evolution of Computer Architecture: Introduction of computer architecture, Elements of Modern Computers, Evolution of Computer Architectures, Classification of parallel computers, System attributes to performance.

Program and Network Properties: Conditions of Parallelism - data and resource dependences, Bernstein's conditions, hardware and software parallelism. Program Flow Mechanisms - control flow versus data flow, data flow architecture, demand driven mechanisms, comparison of flow mechanisms.

Unit-II

System Interconnect Architectures: Network properties and routing, Static connection Networks –Linear Array, Ring & Chordal Ring, Barrel Shifter, Fat Tree, Mesh & Torus, Systolic Arrays, Hypercubes; Dynamic connection Networks – Digital Buses, Switch modules, MINs, Omega-, Baseline-, Crossbar-Network.

Memory Hierarchy Design: Memory hierarchy, Inclusion, coherence & locality; memory capacity planning; Virtual Memory technology – Models, TLB, Paging and Segmentation; Cache Memory Organization - Cache

basics & cache performance, cache addressing models & mapping, multilevel cache hierarchies, interleaved memory.

Unit-III

Multiprocessor and Multicomputer Architectures: Multiprocessor System Interconnects – Hierarchical bus systems, Crossbar Switch and Multiport memory, Multistage and Combining networks; Symmetric shared memory architectures, distributed shared memory architectures, Cache coherence problem, Snoopy cache coherence protocol, directory-based protocols; Multicomputer Generations, Message passing mechanisms – message routing schemes, deadlock and virtual channels, flow control strategies, multicast routing algorithms.

Unit-IV

Overview of Quantum Computing: Qubits, quantum gates, Hilbert spaces, Dirac's notation, Quantum Superposition and Entanglement, Classical computing vs. Quantum computing, Postulates of quantum mechanics, Quantum circuits, quantum parallelism, Quantum circuits, universal gates, Quantum Fourier transform, Shor's factoring algorithm, order finding and periodicity, Grover's quantum search algorithm, Quantum error correcting codes, Quantum cryptography, Applications of Quantum Computing.

Suggested Readings:

- Kai Hwang & Naresh Jotwani: Advanced Computer Architecture; McGraw-Hill.
- Kai Hwang: Advanced computer architecture; TMH.
- D.Sima, T. Fountain, P.Kasuk: Advanced Computer Architecture-A Design space Approach, Addison Wesley.
- M.J Flynn: Computer Architecture, Pipelined and Parallel Processor Design; Narosa Publishing.
- D. A. Patterson and J. L. Hennessey: Computer organization and design, Morgan Kaufmann.

Useful Video links:

Unit No.	Topics	Links
Unit-I	Introduction to Computer Architecture	https://www.nptelvideos.com/lecture.php?id=5631
	Basics of Computer Architecture	https://youtu.be/4TzMyXmzL8M?si=cmIOLt9Fnsyi4OUb
Unit-II	Memory Hierarchy in Computer Organization	https://www.nptelvideos.com/lecture.php?id=5658
	Cache Memory in Computer Architecture	https://www.nptelvideos.com/lecture.php?id=5286
Unit-III	Parallel Processing and Multiprocessing	https://youtu.be/TE4cJqrIHvI?si=UnmelScrq9kRDy_L
	Symmetric and Distributed Memory Architecture	https://youtu.be/WdqdebPmPuQ?si=WQwvPZ9gnUvRcbhS
	Message Passing and Routing in Multiprocessors	https://www.youtube.com/watch?v=T_A4LgEJaIo
Unit-IV	Quantum Computing Overview	https://youtu.be/2SPjEA-4lKk?si=KWUiY4G0mne8CDV8
	Quantum Computing	https://youtu.be/xnmpWfQKPSE?si=gZH7hGpIIdja0wdM

Course Code	**RR-MCA-112A**				
Category	Research Report				
Course Title	**Industry Internship Report/ Project Report/Dissertation–I**				
Scheme and Credits	L	T	P	**Credits**	**Semester-II**
	0	0	6	**3**	
Course Objectives	The objectives of this course are to • Develop effective project planning and management skills. • Enhance technical skills in software development, database design, and system integration. • Apply problem-solving techniques to address technical challenges. • Improve written and oral communication skills. • Understand and apply ethical principles in software development.				
Assessment	50 Marks				
End Semester Examination	50 Marks				
Total Marks	100 Marks				
Duration of Exam	03 Hours				

Course Outcomes: After studying this course, the students will be able to

COs	Skills Demonstrated	RBT Level
CO1	Identify various technologies and fields for practical training.	Level 1: Remember
CO2	Understand the process to make reports and presentation.	Level 2: Understand
CO3	Applying engineering knowledge to solve industrial problems.	Level 3: Apply
CO4	Analyze ethical practices and tools used in different technologies	Level 4: Analyze
CO5	Evaluate the performance on parameters such as communication skills, technical knowledge.	Level 5: Evaluate
CO6	Design and develop the skills to make software/hardware, related to industrial training.	Level 6: Create

Overview:

MCA Dissertation-I is a foundational course that introduces students to the research process in computer science and information technology. This course provides students with the necessary skills and knowledge to undertake independent research projects. Students will learn to identify research problems, conduct literature reviews, design research methodologies, collect and analyze data, and present their findings effectively.

General Guidelines:

Project Selection	Choose a project that is relevant to the field of computer science and information technology. Ensure that the project is feasible and achievable within the given timeframe. Consult with the supervisor to refine the project scope and objectives.
Approval Process	The selected project must be approved by the faculty members/committee appointed by the Head of Department.
Presentation Guidelines	Each student will have 30-40 minutes for their presentation, followed by 5 minutes for Q&A.
Evaluation	The presentation will be evaluated by a committee constituted by the Head of Department. The evaluation will be based on:

Parameters for the Evaluation of Seminar:

Sr. No.	Parameters	Marks Allotted
1	Clarity of the topic	5
2	Project Planning and Management	5
3	Technical Implementation	10
4	Testing and Debugging	10
5	Documentation	10
6	Teamwork and Collaboration	10