School of Engineering and Technology

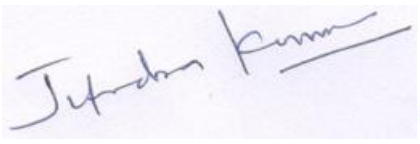
Programme Structure & Syllabus

Master of Computer Application (MCA)

2023-24

K.K. University

Bihar Sharif, Nalanda – 803115



Pro Vice Chancellor
KK University
Berauti, Nepura, Bihar Sharif
Nalanda - 803115 (Bihar)

K. K. UNIVERSITY

BERAUTI, NEPURA, BIHARSHARIF, NALANDA, BIHAR-803115.

School of Engineering & Technology.

Department of Computer Science & Engineering.

Master of Computer Science (MCA)

Objective of the Program:

The objective of the Master of Computer Applications (MCA) program is to prepare students with the necessary knowledge and skills in various aspects of computer applications and information technology. Here are the main objectives of the MCA program:

1. **Technical Skills**: To provide a strong foundation in the core areas of computer science such as programming, data structures, algorithms, database management, networking, and software engineering.
2. **Application Development**: To train students in developing software applications using various programming languages, tools, and technologies.
3. **System Design and Analysis**: To teach principles and techniques for designing and analyzing complex systems and software applications.
4. **Management Skills**: To impart knowledge in areas of management, organization, and project management, especially related to IT projects.
5. **Research Orientation**: To foster research skills and encourage students to undertake research projects in emerging areas of computer science and applications.
6. **Soft Skills**: To develop soft skills such as communication, teamwork, and leadership which are essential for working in the IT industry.
7. **Ethics and Professionalism**: To emphasize ethical responsibilities and professional standards in the practice of computing.
8. **Industry Readiness**: To make students industry-ready by exposing them to real-world problems through internships, projects, and collaborations with industry.

Program Education Outcomes (PEOs):

The Program Education Outcomes (PEOs) of the Master of Computer Applications (MCA) program outline the specific achievements and capabilities that graduates are expected to attain by the time they complete their studies. These outcomes reflect the overarching goals of the program and serve as benchmarks to measure its effectiveness. Typically, the PEOs for an MCA program include:

K.K. UNIVERSITY 803115 NALANDA

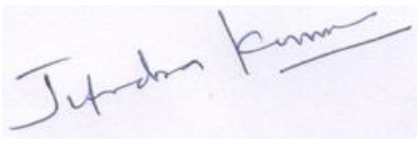
Pro Vice Chancellor
KK University
Berauti, Nepura, Bihar Sharif
Nalanda - 803115 (Bihar)

1. **Technical Competence**: Graduates should demonstrate proficiency in applying knowledge of computing fundamentals, programming languages, algorithms, data structures, database management systems, and software engineering principles to solve complex problems.
2. **Problem-solving Skills**: Graduates should be able to analyze, design, and develop software solutions that meet specified requirements using appropriate methodologies.
3. **Professionalism**: Graduates should exhibit professionalism, ethical behavior, and an understanding of professional, legal, and ethical issues related to computing practices.
4. **Communication Skills**: Graduates should be able to communicate effectively with technical and non-technical audiences through documentation, presentations, and interpersonal interactions.
5. **Teamwork and Leadership**: Graduates should be capable of working effectively as part of a team and demonstrate leadership qualities in various professional and technical contexts.
6. **Continuous Learning**: Graduates should recognize the need for and engage in lifelong learning, including keeping up-to-date with technological advancements and evolving industry trends.
7. **Career Advancement**: Graduates should be prepared for successful careers in the IT industry, academia, research, or entrepreneurship, with the ability to adapt to new technologies and roles over time.
8. **Research Orientation**: Graduates should have the ability to conduct research independently or as part of a team, contributing to the development and advancement of knowledge in computer applications and related fields.

Program Outcomes (POs):

The Program Outcomes (POs) of the Master of Computer Applications (MCA) program outline specific measurable achievements that students are expected to demonstrate upon completion of the program. These outcomes are designed to ensure that graduates have acquired the necessary knowledge, skills, and abilities to succeed in their careers and contribute effectively to the field of computer applications. Here are typical Program Outcomes for an MCA program:

1. **Knowledge**: Graduates should demonstrate a comprehensive understanding of the fundamental principles, theories, and practices of computer science and its applications.
2. **Problem Solving**: Graduates should be able to apply knowledge of computing and mathematics appropriate to the discipline to identify, formulate, and solve complex problems in computer applications.
3. **Design and Development**: Graduates should be able to design, implement, and evaluate computer-based systems, processes, components, or programs to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability.
4. **Tools and Techniques**: Graduates should be proficient in using modern computing tools and techniques necessary for computing practice.
5. **Software Engineering**: Graduates should demonstrate knowledge and understanding of software engineering principles and practices for the design and development of software systems of varying complexity.

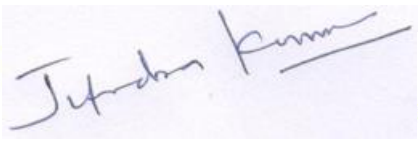



6. **Teamwork**: Graduates should be able to function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline.
7. **Communication**: Graduates should be able to communicate effectively with a range of audiences through written reports, presentations, and discussions.
8. **Professionalism and Ethics**: Graduates should demonstrate an understanding of professional, ethical, legal, security, and social issues and responsibilities.
9. **Lifelong Learning**: Graduates should recognize the need for, and have the ability to engage in, continuing professional development.
10. **Project Management**: Graduates should demonstrate knowledge and understanding of management and business practices related to software projects.

Program Specific Outcomes (PSOs) :

Program Specific Outcomes (PSOs) of the Master of Computer Applications (MCA) program are more detailed and specific than Program Outcomes (POs). They typically focus on the specific competencies and skills that students are expected to achieve in relation to the program's specialized areas or domains. Here are some examples of Program Specific Outcomes for an MCA program:

1. **PSO 1: Application Development**: Graduates should be able to analyze, design, develop, and test software applications using appropriate programming languages, frameworks, and tools.
2. **PSO 2: Database Management**: Graduates should demonstrate proficiency in designing, implementing, and managing databases using relational and non-relational database management systems.
3. **PSO 3: Web Technologies**: Graduates should be able to develop web-based applications using front-end and back-end technologies, and understand principles of web security and performance optimization.
4. **PSO 4: Mobile Computing**: Graduates should be capable of designing and developing mobile applications for various platforms, incorporating usability and user experience principles.
5. **PSO 5: Software Project Management**: Graduates should be able to apply project management principles and practices to plan, execute, and deliver software projects within schedule, budget, and quality constraints.
6. **PSO 6: Data Analytics and Visualization**: Graduates should be proficient in analyzing data, applying statistical techniques, and visualizing findings to support decision-making processes.
7. **PSO 7: Networking and Security**: Graduates should have knowledge of networking protocols, security threats, and measures to ensure secure communication and data protection in computer networks.
8. **PSO 8: Emerging Technologies**: Graduates should be aware of and able to adapt to emerging technologies such as cloud computing, artificial intelligence, blockchain, and Internet of Things (IoT) in the context of computer applications.
9. **PSO 9: Research and Innovation**: Graduates should demonstrate the ability to conduct research, innovate, and contribute to the advancement of knowledge in specialized areas of computer applications through projects, papers, or dissertations.




10. **PSO 10: Entrepreneurship and Innovation**: Graduates should have an entrepreneurial mindset and understand the process of identifying opportunities, developing business models, and launching technology-driven ventures.

These Program Specific Outcomes ensure that graduates of the MCA program are not only proficient in core computer science concepts but also have specialized skills and knowledge relevant to specific domains within the field of computer applications. PSOs provide a clearer focus on the practical application of learning outcomes and prepare students for specialized roles and challenges in the IT industry and related fields.

Master of Computer Application (MCA)

Program/Course Structure

FIRST SEMESTER

S.NO	CODE	COURSE TITLE	CREDIT	L	T	P	Hours per	Internal	External
1	MCAT101	Object Oriented Programming	4	3	2	--	4	30	70
2	MCAT102	Discrete Mathematics	4	4	--	--	4	30	70
3	MCAT103	Problem Solving and Programming	3	3	--	--	3	30	70
4	MCAT104	Statistical Techniques	2	2	--	--	2	30	70
5	MCAT105	Database Management System	4	4	--	--	4	30	70
6	MCAT106	Soft Skills	2	2	--	--	2	30	70
7	MCAL101	Object Oriented Programming Lab	1	--	--	1	2	30	70
8	MCAL103	C Programming Lab	1	--	--	1	2	30	70
9	MCAL105	Database Management System Lab	1	--	--	1	2	30	70
10	MCAL106	Soft Skills Lab	1	--	--	1	2	30	70
TOTAL			**23**	**18**	**2**	**4**	**27**	**300**	**700**

SECOND SEMESTER

S.NO	CODE	COURSE TITLE	CREDIT	L	T	P	Hours per week	Internal Marks	External Marks
1	MCAT20	Core Java	4	3	2	-	4	30	70
2	MCAT20	Data Structure and Algorithm	4	3	1	0	4	30	70
3	MCAT20	Computer Networks	3	3	0	0	3	30	70
4	MCAT20	Dot Net Technology	3	3	0	0	3	30	70
5	MCAT20	Operating System	3	3	1	0	3	30	70

K.K. UNIVERSITY 803115 NALANDA

Pro Vice Chancellor
KK University
Berauti, Nepura, Bihar Sharif
Nalanda - 803115 (Bihar)

6	MCAT20	Accounting & Financial	2	2	0	0	2	30	70
7	MCAL20	Core Java Lab	1	0	0	1	2	30	70
8	MCAL20	Data Structures and Algorithm Lab	1	0	0	1	2	30	70
9	MCAL20	Computer Networks Lab	1	0	0	1	2	30	70
1	MCAL20	Dot Net Lab	1	0	0	1	2	30	70
TOTAL			**23**	**17**	**4**	**4**	**27**	**300**	**700**

K.K. UNIVERSITY 803115 NALANDA

Pro Vice Chancellor
KK University
Berauti, Nepura, Bihar Sharif
Nalanda - 803115 (Bihar)

THIRD SEMESTER

S.NO	CODE	COURSE TITLE	CREDIT	L	T	P	Hours per week	Internal Marks	External Marks
1	MCAT301	Design and Analysis of Algorithms	3	3	1	0	3	30	70
2	MCAT302	Software Engineering	3	3	0	0	3	30	70
3	MCAT303	Advance Java	3	3	2	0	3	30	70
4	MCAT304	Data Mining	3	3	0	0	3	30	70
5	MCAT305	Python Programming	3	3	0	0	3	30	70
6		**Elective I**	4	4	0	0	4	30	70
7	MCAL301	Design and Analysis of Algorithms	1	0	0	4	4	30	70
8	MCAL303	Advance Java Lab	1	0	0	4	4	30	70
9	MCAL305	Python Programming Lab	1	0	0	4	4	30	70
	TOTAL		**22**	**19**	**3**	**12**	**31**	**270**	**630**

Elective I

MCAT306 : **Artificial Intelligence**

MCAT307: Parallel Computing

MCAT308: CYBER Security MCAT309:

Cloud Computing

FOURTH SEMESTER

S.NO	CODE	COURSE TITLE	CREDIT	L	T	P	Hours per week	Internal Marks	External Marks
1	MCAL401	Dissertation (Major Project)	20	0	0	4	20	100	400
2	MCAL402	Seminar	4	0	2	0	4	30	70
TOTAL			**24**	**0**	**2**	**4**	**24**	**130**	**470**

SEMESTER I

Programme Structure	MCA
Subject Code	MCAT101
Course Name	Object Oriented Programming
Course Credits	L(4)+P(1)=5
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**
   This course will introduce the core concepts of object oriented programming And related concepts to the learner. It will also allow him to formulate solutions based On this knowledge and apply them in solving real world problems.

2. **Prerequisite:** To understand the basic computer organization, operating systems and programming

3. **Objective of the Syllabus:**

   - identify and practice the object-oriented programming concepts and techniques, practice the use of C++ classes and class libraries,
   - arrays, vectors, inheritance and file I/O stream concepts.

4. **Course Outcomes:**

S.No.	Course Outcomes(Cos)
CO1	student will understand basics of OOPS and implement these concepts in their programs
CO2	students apply abstract data type concepts and polymorphism to their projects
CO3	inheritance, polymorphism, IO and exception handling is discussed in detail
CO4	students apply generic programming concepts and master STL algorithm in making their programs more effective and industry relevant

5. **Syllabus:**

UNIT I

Introduction to OOPs: What Is Object-Oriented Programming? , Encapsulation, Polymorphism, Inheritance.

C++ Overview: The Origins of C++,The General Form of a C++ Program, different data types, operators, expressions, arrays and strings, reference variables. Function Components, argument passing, inline functions, function overloading, function templates.

UNIT II

Classes & Objects: Introduction, Class Specification, Class Objects, access members, defining member functions, data hiding, constructors, destructors, parameterized constructors, static data members, static member functions, scope resolution operator, Passing Objects to Functions, Returning Objects, Object Assignment.

Pointers and dynamic memory allocation: Pointers, Pointer as function arguments, Dynamic Allocation Operators new and delete, Initializing Allocated Memory, Allocating Arrays, Allocating Objects

UNIT III

Operator overloading: Operator overloading as member functions and using friend functions. Overloading of binary operators like +, -, *.Creating Prefix and Postfix forms of ++, -- Operators, Operator Overloading Restrictions, Operator Overloading Using a Friend Function to Overload ++ or – –, Overloading ( ).

Inheritance: Base Class, Inheritance & protected members, protected base class inheritance, inheriting multiple base classes, Constructors, Destructors & Inheritance. Passing parameters to base Class Constructors, Granting access, Virtual base classes.

UNIT IV

Virtual Functions and Runtime Polymorphism: Virtual function -Calling a Virtual function through a base class reference, Virtual attribute is inherited, Virtual functions are hierarchical, pure virtual functions, abstract classes, using Virtual functions, Early & late binding.

Standard C++ I/O Classes: Old vs. Modern C++ I/O, C++ Streams, The C++ Stream Classes, C++'s Predefined Streams, Formatted I/O, Formatting Using the ios Members, Setting the Format Flags, Clearing Format Flags, Overloading << and >>, manipulators

UNIT V

Exception Handling: Exception Handling, Fundamentals, Catching Class Types, Using Multiple catch Statements, Handling Derived-Class Exceptions, Exception Handling Options, Catching All Exceptions, Restricting Exceptions, Rethrowing an Exception, Understanding terminate( ) and unexpected( ), uncaught_exception( ) Function, The exception and bad_exception Classes, Applying Exception Handling.

STL: Class template, An overview of STL, containers, vectors

TEXTBOOKS:

1. IraPohl,“Object-OrientedProgrammingUsingC++”,PearsonEducationAsia,2003.
2. Object-Oriented Programming with C++ by E Balagurusamy

REFERENCEBOOKS:

1. HerbertSchildt,"CompleteReference",Fourthedition,TMH,2002
2. BjarneStroustrup,“TheC++ProgrammingLanguage”,PearsonEducation,2004.
3. Stanley B. Lippman and Josee Lajoie,“C++Primer”, PearsonEducation,2003.
4. K.R.Venugopal,RajkumarBuyya,T.Ravishankar,"MasteringC++",TMH,2003

OBJECT ORIENTED PROGRAMMING LAB SUB-CODE:MCAL101 CREDIT : 02

Course Objective:

- Gain the basic knowledge on Object Oriented concepts.
- Ability to develop applications using Object Oriented Programming Concepts.
- Ability to implement features of object oriented programming to solve real world problems.

Course Outcomes:

At the end of the course the students are able to:

- To write original code in C++
- To design and solve real world problems.
- To understand relevance of various concepts of object oriented programming

Syllabus:

List of Experiments(one per week):

1 Write a C++ program to find the sum of the given variables using function with default arguments.

2 Write a C++ program to find the value of a number raised to its power using call by value.

3 Write a C++ program to implement the concept of Call by Address.

4 Write a program in C++ to implement the concept of call by reference.

5 Write C++ program to implement inline function.

6 Write a program in C++ to display product detail using classes with array as data members.

7 Write a program in C++ implements the concept of class with constant data member.

8 Write a program in C++ to implement the concept of class with static member functions.

9 Write a C++ program to implement the friend function concept.

10 a) Write a C++ program to implement the concept of unary operator overloading using c++.

b) Write a C++ program to implement the concept of Binary operator overloading.

11 Write a C++ program to implement the concept of Function Overloading.

12 a) To implement single inheritance using c++.

b) To write a C++ program to implement multiple inheritance.

c) To write a C++ program to implement multilevel inheritance.

13 a) Write a C++ program to implement the concept of class template.

b) Write a C++ program for swapping two values using function templates

REFERENCE BOOKS:

1. An Introduction to Operating Systems, P.C.P Bhatt, 2nd edition, PHI, *Year*.

2. Modern Operating Systems, Andrew S Tanenbaum, 3rd Edition, PHI, *Year.*

Programme Structure	MCA
Subject Code	MCAT102
Course Name	Discrete Mathematics
Course Credits	4(L)=4
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**

   This course will introduce the core concepts of discreet mathematical structures ,

   Their applications , and problem solving to the learner.

2. **Prerequisite:** To understand the basic mathematical concepts , mathematical analysis, set theory.

3. **Objective of the Syllabus:**

   The primary objective of the course is that students should learn a particular set of mathematical facts and how to apply them. In particular it teaches students how to think logically and mathematically through five important themes: mathematical reasoning, combinatorial analysis, discrete structures, algorithmic thinking, and applications and modeling. A successful discrete mathematics course should carefully blend and balance all five themes.

4. **Course Outcomes:**

S.No.	CourseOutcomes(Cos)
CO1	student will study formal logic, basic principle of DM like permutation, combination, sets, relations which will help students in solving both simple and complex mathematical problems.
CO2	students study Boolean algebra, lattices, well ordering recursive definition etc. which are required for better understanding of working with well-ordered numbers
CO3	Students understand RSA cryptosystems and error correcting codes that will make them understand how cryptography in computer works using DM
CO4	students study graphs and trees enabling them to organize data in a well-defined format

5. **Syllabus:**

UNITI

Sets:Definitionandtypes,Setoperations,Partitionofset,Cardinality(Inclusion-Exclusion&AdditionPrinciples),Recursivedefinitionofset.Functions:Concept,SomeSpecialFunctions(Polynomial, Exponential & Logarithmic, Absolute Value, Floor & Ceiling, Mod &DivFunctions),PropertiesofFunctions,CardinalityofInfiniteSet,Countable&UncountableSets,thePigeonhole&Generalized Pigeonhole Principles, Composition of Functions.

UNITII

Relations: Boolean Matrices, Binary Relation, Adjacency Matrix of Relation, Properties ofRelations,OperationsonRelations,TheConnectivityRelations,TransitiveClosure-Warshall'sAlgorithm,Equivalencerelations-CongruenceRelations,EquivalenceClass,NumberofPartitions of a Finite Set,Partial &Total Orderings.

UNITIII

ProofMethods:Vacuous,Trivial,Direct,IndirectbyContrapositiveandContradiction,Constructive&Non-constructiveproof,Counterexample.TheDivisionAlgorithm,Divisibility Properties (Prime Numbers & Composite Numbers), Principle of MathematicalInduction,theSecondPrincipleofMathematicalInduction,FundamentalTheoremofArithmetic. Algorithm Correctness: Partial Correctness, Loop Invariant. Testing the partialCorrectnessoflinear&binarysearch, bubble&selection sorting.

UNITIV

Graph Theory: Graphs – Directed, Undirected, Simple,. Adjacency & Incidence, Degree of Vertex, Sub graph, Complete graph, Cycle & Wheel Graph, Bipartite & Complete Bi partite Graph, Weighed Graph, Union of Simple Graphs. CompleteGraphs. Isomorphic Graphs, Path, Cycles & CircuitsEuclerian & Hamiltonian Graphs. Planar Graph: Kuratowski's Two Graphs, Euler's Formula, Kuratowski's Theorem. Trees: Spanning trees- Kruskal's Algo, Finding Spanning Tree using Depth First Search, Breadth First Search, Complexity of Graph, Minimal Spanning Tree.

UNITV

LanguageofLogic:Proposition,CompoundProposition,Conjunction,Disjunction,Implication,Converse,Inverse&Contrpositive,BiconditionalStatements,tautology,Contradiction &Contingency,Logical Equivalences,Quantifiers, Arguments.

References:

a. DiscreteMathematicswithApplications,Koshy,ELSEVIER
b. DiscreteMathematicalStructuresByLipschutz&Lipson,TMH
c. DiscreteMathematicalStructures,Kolmanet.al,Pearson

Programme Structure	MCA
Subject Code	MCAT103
Course Name	Problem Solving and Programming in C
Course Credits	3 (L)+1 (P)=4
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**

   This course will introduce the core concepts of C programming language, Data structures , files and how to write correct code to solve real life problems using The C programming language.

2. **Prerequisite:** To understand the basic computer organization, operating system structures, processes and threads.

3. **Objective of the Syllabus:**
   - Programming basics and the fundamentals of C
   - Data types in C
   - Mathematical and logical operations
   - Using if statement and loops
   - Arranging data in arrays
   - Implementing pointers
   - File management and dynamic memory allocation

4. **Course Outcomes:**

S.No.	CourseOutcomes(Cos)
CO1	the students will be able to understand the basics of C language, commands and data structure. This will help them in understanding basics of programming languages
CO2	students will study advanced data types, pointers and dynamic memory allocation, develop an application by managing memory.
CO3	the students will be taught about C preprocessors, using Linux with C, string and file Handling. Develop macros & use it in an application development
CO4	students will study Linux admin and other commands. They will know how to create and manage multiple users on Linux systems.

5. **Syllabus:**

UNIT I

Problem Solving: Problem Identification , Analysis , Flowcharts, Decision tables, Pseudocodesandalgorithms,Programcoding,ProgramTestingandexecution;Typesofprogramminglanguages,Translators, Interpreters, Compilers , Assemblers and their comparison. Fundamentals of C language: History of C Language, Structure of a C program, Variables, Constants, Keywords, Data types, Operators, Expressions and their evaluation using rules of hierarchy, typecasting, Input/ output statements, Assignment statements, Control statements: if-else, switch, while, do-while, for, nested loops, break, continue, go to statements.

UNIT II

Functions: Declaration, Definition, function prototype, passing arguments: call by value, call by reference, Recursion and stack, Use of library functions, adding functions to the library, Functions with variable arguments; Storage classes: automatic, external and static variables. Arrays: Defining and processing arrays, Passing array to a function, Using multi-dimensional arrays, Solving matrices problems using arrays;

UNIT III

Strings and Pointers: String: declaration, Operations on strings, Two-dimensional array of characters; Pointer: declaration, Operations on pointers, Passing pointers to functions, Arrays of pointers, Array of pointers to strings. Structure and Union: Structures: Defining and processing, Passing structure to a function, Arrays of structures, Pointers and structures, Uses of structures; Unions: Defining and processing, Pointers and union, Union of structures, Uses of union.

UNIT IV

Files Handling: Concept of files, file opening modes, opening and closing of a file, reading from a file, writing on to a file, Error handling during I/O operations, Record /In Files. Miscellaneous: Command line arguments, Enumerated data types, Renaming data types with typedef, Pre-processor directives, Using GCC for programming In C under Linux.

Suggested Readings:

1. Kanetkar, Yashavant, 2016, Let UsC,BPBPublications.
2. Cooper,Mullish,1987:TheSpiritofC,AnIntroductiontoModernProgramming,JaicoPubl.House, New Delhi.
3. Kenneth,A.:CProblemSolvingandProgramming,PrenticeHall International.
4. Kerninghan,B.W.&Ritchie,D.M.:The C Programming Language, Prentice Hall of International.
5. Gottfried,B.:TheoryandProblems ofProgramminginC, SchaumSeries.
6. Jones,A./KenithHarvow:CProgrammingwithProblemSolving,WileyIndiaPvt. Ltd.
7. Gookin,Dan:CProgramming,WileyIndiaPvt.Ltd

C PROGRAMMING LAB SUB-CODE:MCAL103 CREDIT: 02

Course Objective:

Understand programming basics and get a holistic view of the C Programming language, detailing all the aspects of the C language from data types, to operators and expressions, to if statements, further to loops, arrays, strings and pointers. Also have hands-on training to help write and test code , and prepare for real-life application.

Course Outcomes:

- Develop a C program
- Control the sequence of the program and give logical outputs
- Implement strings in your C program
- Store different data types in the same memory
- Manage I/O operations in your C program
- Repeat the sequence of instructions and points for a memory location
- Apply code reusability with functions and pointers
- Understand the basics of file handling mechanisms
- Explain the uses of pre-processors and various memory models

Syllabus:

Lab List of Experiments in C:(one per week)

1 Write a C program to find the sum of individual digits of a positive integer.

2 Write a C program to generate Fibonacci series.

3 Write a C program to generate all the prime numbers between 1 and n is a value supplied by the user.

4 Write a C program to find the roots of a quadratic equation.

5 Two integer operands and one operator form user, performs the operation and then prints the result.

6 Write a C program to find the factorial of a given integer by using recursive and non-recursive functions.

7 A C program to find both the largest and smallest number in list of integers

8 Write A C- Program To Determine If The Given String Is A Palindrome Or Not

9 Example of Array In C programming to find out the average of 4 integers

10 Write a program in c to Addition of two matrix in C

11 Write a C program to Implement the following searching method. i) linear search ii) Binary search

12 Write C programs that implement the following sorting methods to sort a given
list of integers in ascending order by using Bubble sort.

Programme Structure	MCA
Subject Code	MCAT104
Course Name	Statistical Techniques
Course Credits	2(L)=2
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**

   This course will introduce the core concepts of statistical techniques, their applications , uses and applicability in real world situations.

2. **Prerequisite:** To understand the basic mathematics, statistical concepts and probability.

3. **Objective of the Syllabus:**

   Students will summarize data visually and numerically. Students will build and assess data-based models. Students will learn and apply the tools of formal inference.

4. **Course Outcomes:**

S.No.	CourseOutcomes(Cos)
CO1	calculate and apply all measures of location and measures of dispersion for grouped and ungrouped data.
CO2	Student can apply discrete and continuous probability distributions to all of business problems.
CO3	3.1 Student can perform all test of Hypothesis 3.2 Student can calculate all confidence interval for a population
CO4	Student knows all parts of nonparametric test such as the ChiSquare test for Independence as well as Goodness of Fit.
CO5	

5. Syllabus:

UNITI

Probability and probability distribution Various definition of probability, additive and multiplicative theorem, independent event, probability distribution, mathematical expectations, additive and multiplicative theorem of expectation ,Binomial, Poisson and normal distribution, Fitting of Binomial distribution.

UNITII

DescriptiveStatistics

Measure of central tendencies, Dispersion , Measure of Dispersion, product of moments,correlationcoefficients,Rankcorrelation,LinearRegression,Propertiesofregressionc oefficient,Multiple linear regression.

UNITIII

NumericalMethods

Transcendentalandpolynomialequations:Iterativemethods,Regula-Falsimethod,-Raphsonmethod.Rootsofpolynomial:GiraffesandBairstowmethods.Solutionofsystemofalgeb raiclinearequations:Gausselimination,GaussJordan,Data fitting,Methodof least square.

UNITIV

Testof significance

Nullandalternativehypothesis,Onetailandtwotailtests,Twotypesoferrors,Largesampletests, Small sample tests, Test of single mean, Test of equality of two means, Paired test, testofgoodness of fit, testofindependenceof attributes,Test of variance.

UNITV

Sampling techniques and analysis of variance

Sampling and complete enumeration, Simple random Sampling, Stratified random Sampling, Proportional and Optimum allocations.

ANOVA: One way and two way classifications.

TEXT AND REFERENCE BOOKS:

1. G.W.SnedcorandW.G.Cochrass:“StatisticalMethods”,6EdnEastWestpress.
2. S.C.Gupta:“IntroductiontoMathematicalStatistics”,1973,SultanChand.
3. S.C. Chapra and R. P.Canales:“NumericalMethodsforEngineers”,2002TMH.

Programme Structure	MCA
Subject Code	MCAT105
Course Name	Database Management System
Course Credits	4(L)+1(P)=5
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**
   The objective of the course is to present an introduction to database management systems, with an emphasis on how to organize, maintain and retrieve - efficiently, and effectively - information from a DBMS

2. **Prerequisite:** To understand the basic computer organization, operating system and programming.

3. **Objective of the Syllabus:**
   - Describe the fundamental elements of relational database management systems
   - Explain the basic concepts of relational data model, entity-relationship model, relational database design, relational algebra and SQL.
   - Design ER-models to represent simple database application scenarios
   - Convert the ER-model to relational tables, populate relational database and formulate SQL queries on data.
   - Improve the database design by normalization..

4. **Course Outcomes:**

S.No.	CourseOutcomes(Cos)
CO1	intend to make the students understand the importance and role of DBMS in project development.
CO2	discuss SQL queries and views with the intention to insert, update, delete and extract an appropriate data to/ from the database.
CO3	students will be able to apply the knowledge about oracle DBMS in the real world environment
CO4	student will be able to interpret the concept of normalization, concurrency control and recovery both in centralized and distributed database environment
CO5	

5. **Syllabus:**

UNIT– I

Introduction;Anexample;CharacteristicsofDatabaseapproach;Actorsonthescreen;Advantages of using DBMS approach; A brief history of database applications; Data models,schemasandinstances;Three-schemaarchitectureanddataindependence;Databaselanguagesand interfaces; The database system environment; Centralized and client-server architectures; Classification of Database Management systems, ACID Properties.

UNIT– II

DataModels-Introduction,DataAssociation-(Entities,AttributesAndAssociations,Relationship Among Entities, Representation Of Association And Relationship), Data Model Classification-(Approaches To The Relational Model, Hierarchical Model & Network ModelWithanExamples),Entity–RelationshipModel,EntitySets,AttributesandKeys;Relationship types, Relationship Sets, Roles and Structural Constraints; Weak Entity Types; Refining the ER Design; ER Diagrams, Naming Conventions and Design Issues;

UNIT-III

Relational Model Concepts; Relational Model Constraints and Relational Database Schemas;KeysinDBMSUpdateOperations,Transactionsanddealingwithconstraintviolations;UnaryRelational Operations: SELECT and PROJECT; Relational Algebra Operations from SetTheory;BinaryRelationalOperations:JOINandDIVISION;AdditionalRelationalOperations; Examples of Queries in Relational Algebra; Relational Database Design UsingER-to-Relational Mapping.

UNIT-IV

Normalization, First –Second- Third Normal Forms, Relational With More Than One Candidates Key ,Good And Bad Decomposition, Multivalued Dependency, Fourth Normal Form, Fifth Normal Form. Network Data Model, Hierarchical Data Model: The Tree Concept, Data Manipulation.

UNIT– V

TheACIDProperties;TransactionsandSchedules;ConcurrencyControl:ProblemOfConcurrent Access, Resource Looking, Deadlock, Database Recovery: Restore, Backward & Forward Recovery, Introduction to crash recovery.

DATABASE MANAGEMENT SYSTEMS LAB SUB-CODE:MCAL105 CREDIT: 02

AIM: The aim of this laboratory is to inculcate the abilities of applying the principles of the database management systems. This course aims to prepare the students for projects where a proper implementation of databases will be required**.**

Objectives of the course:

- To understand data definitions and data manipulation commands
- To learn the use of nested and join queries
- To understand functions , procedures and procedural extensions of databases
- To be familiar with the use of a frontend tool
- To understand design and implementation of typical database

applications

LISTOFPROGRAMS:

1. DataDefinitionCommands,DataManipulationCommandsforinserting,deleting,updating And retrieving Tables and Transaction Control statements
2. Database Querying– Simple queries , Nested queries , Subqueries and Joins
3. Views , Sequences ,Synonyms
4. Database Programming: Implicit and Explicit Cursors
5. Procedures and Functions
6. Triggers
7. Exception Handling
8. Database Design using ER modeling , normalization and Implementation for any application
9. Database Connectivity with Front End Tools
10. Case Study using real life database applications

Programme Structure	MCA
Subject Code	MCAT106
Course Name	Soft Skills
Course Credits	2(L)+1 (P)=3
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**
   This course will introduce the core concepts of communication and soft skills to the learner. It will also allow him to understand common mistakes and how to effectively put forward one's views while working in a team.

2. **Prerequisite:** To understand the basic English, pronunciation , grammar .

3. **Objective of the Syllabus:**
   - To encourage the all round development of students by focusing on soft skills.
   - To make the engineering students aware of the importance, the role and the content of soft skills through instruction, knowledge acquisition, demonstration and practice.
   - To develop and nurture the soft skills of the students through individual and group activities.
   - To expose students to right attitudinal and behavioral aspects and to build the same through activities

4. **Course Outcomes:**

S.No.	CourseOutcomes(Cos)
CO1	Effectively communicate through verbal/oral communication and improve the listening skills
CO2	Actively participate in group discussion / meetings / interviews and prepare & deliver presentations
CO3	Become more effective individual through goal/target setting, self motivation and practicing creative thinking.
CO4	Function effectively in multi-disciplinary and heterogeneous teams through the knowledge of team work, Inter-personal relationships, conflict management and leadership quality.
CO5	

5. **Syllabus:**

UNIT I: Self Awareness and self-Development

Self-Assessment, Self-Awareness, Perceptions and Attitudes, Positive Attitude, Values and Belief Systems, Self-Esteem Self-appraisal, Personal Goal setting, Career Planning, Personal success factors, Handling failure, Emotional Intelligence, Lateral thinking, Depression and Habit, relating SWOT analysis & goal setting ,prioritization.

UNITII: Communication Skill

Importance of communication, Aspects of communication, communication through words,communicationthroughbodylanguage,communicationthroughtechnology,Oral communication, Listening Skills, Group Discussion and Interview Skills, Presentation skills: preparing the presentation, performing the presentation,
Written communication: Reading comprehension, précis writing, Business and technical reports, Styles, Business correspondence, Memorandum writing, Notice, Agenda and Minutes, Research papers and articles, Advertising and job Description, Mechanics of Manuscript preparation.

UNIT III : Interpersonal relationship
Teamwork, effectiveness, Group discussion, Decision making- Team Communication. Team,ConflictResolution,TeamGoalSetting,TeamMotivationUnderstanding Team Development, Team Problem Solving, Building the team dynamics , Multicultural Diversity and Socializing

UNIT IV: Leadership Skills

Leaders:theirskills,roles,andresponsibilities.Vision,Empoweringanddelegation,motivatingothers, organizational skills, team building, Organizing and conducting meetings, decision making ,giving support, Vision, Mission, Coaching, Mentoring and counseling, Appraisals and feedback , conflict , Power and Politics , Public Speaking.

UNIT V:

OtherSkills

Managing Time, Managing Stress, Meditation. Improving personal memory, Study skills that include Rapid Reading, Notes Taking, Self-learning, Complex problem solving and creativity, listening skillsandspeakingskills,CorporateandBusinessEtiquettes.

UNIT VI: Ethics in Engineering Practice and Research

Introduction to ethical reasoning and engineer ethics, Right and responsibilities regarding Intellectual property, work place right sand responsibilities, Central Professional Responsibilities of Engineers, Responsibility for environment. Practical's to be done:- **TermWork/ Assignments SWOT Analysis** Personal & Career Goal setting – Short term & Long term Presentation Skill Dining Etiquettes Letter/Application/Notice/Agenda/MinuteswritingReportwritingListeningskills using Language laboratory Group discussion Resume writing Team Activity Public Speaking

Reference Books:

1. DevelopingCommunicationSkill:KrishnaMohan,MeeraBanerji,-MacMillanIndiaLtd.
2. BNGhosh,:ManagingSoftSkillsforPersanalityDevelopment"McGrawHill
3. EthicsinEngineeringPracticeandResearch:CarolineWhitbeck,CambridgeUniversitypress
4. ACourseInCommunicationSkills:KiranmaiDutt,CambridgeUniversitypress
5. EnglishforBusinessCommunication:SimonSweeney,CambridgeUniversityPress
6. BasicsOfCommunicationInEnglish:FrancisSounderaj,MacMillanIndiaLtd.
7. GroupDiscussionsandInterviewSkills:PriyadarshiPatnaik,CambridgeUniversityPress
8. Professional Presentations: MalcolmGoodale, CambridgeUniversityPress
9. AnIntroductiontoProfessional EnglishAndSoftSkills:Das,CambridgeUniversityPress
10. ApracticalcourseinEffectiveEnglishspeakingskills,G.K.Gangal,PHIPublication

K. K. UNIVERSITY

BERAUTI, NEPURA, BIHARSHARIF, NALANDA, BIHAR-803115.

School of Engineering & Technology.

Department of Computer Science & Engineering.

SEMESTER II

Programme Structure	**MCA**
Subject Code	**MCAT 201**
Course Name	**Core Java**
Course Credits	**4 (L) +1(P)=5**
Total Course Credit	**92**

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**
   This course provides basic Java training through a carefully structured curriculum that starts with an introduction to various. As participants progress, they will gain hands-on experience with Java's syntax and class structures, reinforcing OOP concepts and mastering control statements, data types, and methods.

2. **Prerequisite:** To understand the basic computer software ,editors ,programming concepts

3. **Objective of the Syllabus:**

   To identify Java language components and how they work together in applications. To design and program stand-alone Java applications. To learn how to design a graphical user interface (GUI) with Java Swing. To understand how to use Java APIs for program development.

4. **Course Outcomes:**

S.No.	CourseOutcomes(Cos)
CO1	understand the basics of JVM and java programming language using various programs
CO2	the concept of exception handling, multithreading, input output programming and N/W programming is illustrated in real life projects
CO3	the concepts of Event handling, collection and JDBC are applied in various programs so that the students understand the concept well.
CO4	students are made to experiment on RMI, java beans and swings.

CO5	

5. **Syllabus:**

UNIT-I

Basics of Java : History and Basics of Java, Java Environment, JDK Tools, Java VirtualMachine, Java Program Structure, Java Language- Tokens, Keywords, Constants, Variables,and Data Types. Operators and Expressions, Statements - Decision Making, Branching andLooping,LabeledLoopsStatement,JumpStatements:Break,Continue,andReturn,CommandLineArgument.

UNIT-II

Classes and Objects: Classes, Objects, Defining a Class, Adding Variables and Methods,CreatingObjects,AccessingClassMembers,Constructors,StaticMembers,NestingofMethods, Inheritance and Polymorphism: Basics Types, Extending a Class, Using Super,MethodOverloading,MethodOverriding,FinalVariablesandMethods,FinalClasses,FinalizeMethod,Abstract Methods and Classes, VisibilityControl.

UNIT-III

One and Two Dimension Arrays, String Array, String and String Buffer Classes, Vectors,WrapperClasses.Interfaces:DefiningInterfaces,ExtendingInterfaces,ImplementingInterfaces, Accessing Interface Variables, Packages: System Packages, Naming Conventions,CreatingPackages,AccessingaPackage,UsingPackage,AddingaClasstoaPackage,HidingClasses.ExceptionHandling:IntroductiontoExceptionHandling,Try-Catch,Finally,Throws,JavaThread Model:Life Cycleof aThread, ThreadClass, Runnable Interface

UNIT-IV

AppletProgramming:CreatingandExecutingJavaApplets,InsertingAppletsinaWebPage,Applet Tag, Local and Remote Applets, Applets Vs. Applications, Applets Life Cycle. AWTClasses,SwingClasses,EventHandling,AWTProgramming:WorkingwithWindows,Graphics and Text, Using AWT Controls, Layout Managers and Menus, Handling Image,Animation, Sound and Video. Java Swing: Japplet, Icons and Labels, Text Fields, Buttons,Radio Buttons, Check Boxes, Combo Boxes, List Boxes, Tabbed and Scroll Panes, Tables.EventHandling:

UNIT-V

I/OStream:IntroductionofI/OStream,TypesofStreams,StreamClassHierarchy,UsingFileClass,ByteStreamsVsCharacterStreams,TextfileVs.BinaryFile,StandardI/OStreams,andRandomAccessFile,Serialization.JDBC, JDBC Drivers, Types of JDBC Drivers, Connecting with Database. J2EE:

Introduction of J2EE, Web Application Basics, Architecture and Challenges of Web Application, Servlet, Servlet Life Cycle, Developing and Deploying Servlets.

ReferenceBooks:

1. E.Balagurusamy,"ProgrammingwithJava,aPrimer",TMH,ISBN-13:978-0-07-061713-
2. PatrickNaughtonandHerbertSchildt,"Java:theCompleteReference", TMHPublication.
3. Yashavantkanetkar,"LetusJava",BPBPublications.
5. CayHorstmann, "BigJava", WileyPublication
6. PeterNorton, "JavaProgramming",TechmediaPublications.
7. JosephWeber,"Using Java1.2", PHI

CORE JAVA LAB SUB-CODE:MCAL201 CREDIT: 02

Course Objective:

- To understand the functionalities of various concepts of Object Oriented programming
- To understand various data types and language functionalities of Java
- To be able to use Java for solving real-life problems

Course Outcomes:

At the end of the course the students are able to:

- Able to solve real world problems using OOP techniques.
- Able to understand the use of abstract classes.
- Able to solve problems using java collection framework and I/o classes.
- Able to develop multithreaded applications with synchronization.
- Able to develop applets for web applications.
- Able to design GUI based applications

Syllabus:

List of Experiments in Java (about one per week)

1 Write a java program to find the Fibonacci series using recursive and non-recursive functions.
2 Write a java program to multiply two given matrices.
3 Write a java program for Method overloading and Constructor overloading.
4 Write a java program to display the employee details using Scanner class.
5 Write a java program that checks whether a given string is palindrome or not.
6 Write a java program to represent Abstract class with example.
7 Write a java program to implement Interface using extends keyword.
8 Write a java program to create inner classes.
9 Write a java program for creating multiple catch blocks.
10 Write a Java program that implements a multi-thread application that has three threads.
11 Write an applet program that displays a simple message.
12 Write a Java program compute factorial value using Applet.
13 Write a program for passing parameters using Applet.
14 Write a java program for handling Mouse events and Key events.
15 Write a java program that connects to a database using JDBC.
16 Write a java program that works as a simple calculator. Use a Grid Layout to arrange
Buttons for digits and for the + - * % operations. Add a text field to display the result.

Programme Structure	MCA
Subject Code	MCAT 202
Course Name	Data Structure and Algorithms
Course Credits	4 (T) +1 (P)=5
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**

   This course will introduce the concepts of data structures and how to design, Analyze and use various algorithms related to various data structures.

2. **Prerequisite:** To understand programming in C and C++

3. **Objective of the Syllabus:**

   1. To provide the knowledge of basic data structures and their implementations.
   2. To understand importance of data structures in context of writing efficient programs.
   3. To develop skills to apply appropriate data structures in problem solving

4. **Course Outcomes:**

S.No.	CourseOutcomes(Cos)
CO1	Ability to program data structures and use them in implementations of abstract data types
CO2	Ability to devise novel solutions to small scale programming challenges involving data structures and and recursion.
CO3	Understanding of basic algorithmic complexity
CO4	Gain practical knowledge of how to implement various algorithms in day to day situations
CO5	

5. Syllabus:

UNIT-1

Introduction, classification, memory allocation, algorithm analysis, data structure operations,recursion,abstract datatype(ADT).

UNIT II

Datastructurerepresentation,staticanddynamicprogramming,Array,Arrayaddresscalculation,sparsematrix,Linkedlistandtheiroperations,singlylinkedlist,doublylinkedlist,circularlinked list, polynomials.

UNIT III

Linear Data structure, stack terminology, operation on stack, push and pop, applications of stack, infix, prefix and postfix notations, conversion and examples, Queue and its implementation, operations, types of queue, applications of queue.

UNIT IV

Non-linear data structure – Representation of graph, Graph Traversals - Depth-first traversal,breadth-firsttraversal,applicationsofgraphs,Topologicalsort,shortest-path algorithms, minimum spanning tree – Dijkstra's, Prim's and Kruskal's algorithms, Tree introduction, tree representation, binary tree and its creation, tree traversal, binary search tree, B Tree, B+ tree, extended binary tree, Huffman algorithm coding.

UNIT V

Sorting techniques, Bubble, selection, insertion, quick, merge, heap sort, searching, linear andbinary search and hashing,

REFERENCES:

1. M. A. Weiss, “Data Structures and Algorithm Analysis in C++”, Pearson Education Asia,2013. 2. Tanaenbaum A.S.,Langram Y. Augestein M.J “ Data Structures using C” PearsonEducation, 2004

3. AnanyLevitin “Introduction to the Design and Analysis of Algorithms” Pearson Education2003.

4. E. Horowitz, S.Sahni and Dinesh Mehta,“Fundamentalsof Data structures in C++”,UniversityPress, 2007.

5. E. Horowitz, S. Sahni and S. Rajasekaran, “Computer Algorithms/C++”, Second Edition,UniversityPress, 2007.

6. ReemaThareja,“DataStructuresusingC”, OxfordPress,2012.

7. V. Aho, J. E. Hopcroft, and J. D. Ullman, “Data Structures and Algorithms”, PearsonEducation,1983. 8. T. H.Cormen,

DATA STRUCTURES LAB SUB-CODE:MCAL 202 CREDIT: 02

Objectives of the course:

1. To implement linear and non-linear data structures
2. To understand the different operations of search trees
3. To implement graph traversal algorithms
4. To get familiarized to sorting and searching algorithms

Course Outcomes:

At the end of the course the students are able to:

1. To understand basic data structures, their implementation and some of their standard applications.
2. To develop the ability to design and analyze basic algorithms and prove their correctness using the appropriate data structure learned in the course.

Syllabus:

LIST OF EXPERIMENTS(about one per week)

1 Array Implementation of List ADTs
2 Linked List Implementation of List ADTs
3 Array Implementation of Stack ADTs
4 Linked List Implementation of Stack ADTs
5 Array Implementation of Queue ADTs
6 Linked List Implementation of Queue ADTs
7 Linear Search
8 Binary Search
9 Bubble Sort
10 Insertion Sort

Programme Structure	MCA
Subject Code	MCAT 203
Course Name	Computer Networks
Course Credits	3 (T) +1 (P)=4
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**

This course will introduce the core concepts of operating systems, such as processes and threads, scheduling, synchronization, memory management, file systems, input and output device management and security. The goal of the programming assignments is to give students some exposure to operating system code.

2. **Prerequisite:** Basic knowledge of programming, Data Structures and Object Orient Programming C

3. **Objective of the Syllabus:**

1. Build an understanding of the fundamental concepts of computer networking.

2. Familiarize the student with the basic taxonomy and terminology of the computer networking area.

3. Introduce the student to advanced networking concepts, preparing the student for entry Advanced courses in computer networking.

4. Allow the student to gain expertise in some specific areas of networking such as the design and maintenance of individual networks.

4. **Course Outcomes:**

S.No.	Course Outcomes(Cos)
CO1	introduction of networking, OSI reference model, and digitization.
CO2	the concept of data link layer and the ALOHA protocols has been elaborated to the students with simulations.
CO3	detail of Network layer along with distinction between IPv4 and IPv6
CO4	elaborated the use of Transport layer and Application layer in OSI model.
CO5	

5. **Syllabus:**

UNIT 1

Introduction to Computer Network: Definition and uses of Computer Network, Criteria fora Data Communication Network, Classification of Computer Network, Network Architecture,OSIReferenceModel,DataCommunicationConceptsandterminology,TransmissionImpairments,Transmission Medium.

UNITII

DataEncoding:LineEncoding,TypesofLineCoding,Analog–to-DigitalConversion,TransmissionModes, Error Detection and Correction, Framing,Flow and Error Control

UNIT III

Error Correction and Detection Protocols: Protocols for Noiseless Channels, Protocols forNoisy Channels, High-level Data Link Control Protocol (HDLC), Point-to-Point Protocol(PPP),MAC andLLC Sub-layer.

UNITIV

Wired LANs Ethernet: IEEE 802 Standards, Standard Ethernet, Changes in the Standard,Fast Ethernet, Gigabit Ethernet, Routing Characteristics, Routing Algorithms, Comparison ofRoutingAlgorithms,Broadcast Routing,MulticastRouting, RoutinginAd HocNetworks.

UNITV

Network Layer: IP address Scheme, Subnet Addressing, Subnet Masks, IPV4 Addressing,IPV6Addressing,AddressResolutionProtocol(ARP),andReverseAddressResolutionProtocol(RARP).ServicesofTransportLayer,ConnectionEstablishment,ConnectionRelease, Transport Layer Protocols, Congestion, Simple Mail Transfer Protocol (SMTP), andSendmail Tool, FileTransfer Protocol (FTP), Telnet Protocol.

UNITVI

InternetandWWW:InternetBasics,HypertextTransferProtocol,WorldWideWeb(WWW),SecurityinInternet, E-mail Security.

SuggestedReferences:

1. KuroseandRose–"Computernetworking-Atopdownapproachfeaturingtheinternet"–PearsonEducation
2. S. Keshav,An engineering approachtocomputer networking:ATMnetworks,the

internet,andthe telephonenetwork, Addison-Wesley, 1997.
3. L.L.PetersonandB.S.Davie,Computernetworks:asystemsapproach,3rdEdition,Morgan
KaufmannPublishers,2001.

COMPUTER NETWORKS LAB SUB-CODE:MCAL203 CREDIT: 02

Course Objective:

1. To understand the functionalities of various layers of OSI model
2. To explain the difference between hardware, software; operating systems, programs and files.
3. Identify the purpose of different software applications.

Course Outcomes:

At the end of the course the students are able to:

1. Understand and explain Data Communications System and its components.
2. Identify the different types of network topologies and protocols.
3. Enumerate the layers of the OSI model and TCP/IP. Explain the function(s) of each layer.
4. Identify the different types of network devices and their functions within a network
5. Understand and building the skills of subnetting and routing mechanisms.
6. Familiarity with the basic protocols of computer networks, and how they can be used to assist in network design and implementation

LISTOFEXPERIMENT(about one per week)

1 Study of different types of Network cables and practicallyimplement the cross-wired cable and straight through cable usingclamping tool.
2 Study of Network Devices in Detail.
3 Study of network IP.
4 Connect the computers in Local Area Network.
5 Study of basic network command and Network configurationcommands.
6 Configure a Network topology using packet tracer software.
7 Configure a Network topology using packet tracer software.
8 Configure a Network using Distance Vector Routing protocol.
9 Configure Network using Link State Vector Routing protocol.

Programme Structure	MCA
Subject Code	MCAT 204
Course Name	Dot Net Technology
Course Credits	3 (T) +1 (P)=4
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**
   Through this . NET programming course, participants will delve into object-oriented programming techniques, manage data types and conversions, and control program execution flows. The curriculum extends to web development with ASP.NET, teaching students to construct dynamic websites and process web forms.

2. **Prerequisite:** To understand the basic computer organization, operating system structures, processes and threads.

3. **Objective of the Syllabus:**
   To enable the students to
   - To learn about basic features of .NET and its controls
   - To create an ASP.NET application using standard .NET Controls
   - To learn about connecting data sources using ADO.NET and managing them.
4. **Course Outcomes:**

S.No.	CourseOutcomes(Cos)
CO1	discussed the basic concept of CLI, CTS, CLS, CLR and CIL to students along with the concept of assemblies.
CO2	basics of C# programming have been introduced to students with the help of small programs and projects
CO3	advance features of C# programming like delegates and events, collections have been discussed.
CO4	Introduce the concept of ADO.NET and demonstrate the creation and working of .NET application.
CO5	Explain the concept of LINQ, LINK to SQL and extension methods with the use of database linking in projects.

5. **Syllabus:**

UNIT-I

Introduction to .Net, .Net Framework Features & Architecture, CL, Common Type System,MSIL,Assemblies:TypesofAssemblies,ClassLibraries.EventDriveProgramming,Methodsand Events Related with Mouse and Keyboard. Programming into Visual Studio, Types ofProject in .Net, IDE of VB.Net- Menu Bar, Toolbar, Project Explorer, Toolbox, PropertiesWindow,Form Designer, FormLayout,ImmediateWindow, ASP&HTMLForms

UNIT-II

the VB.Net Language- Variables, Declaring Variables, Data Types, Scope & Lifetime of aVariable, Arrays, Types of Array, Control Array, Subroutine, Functions, Passing Argument toFunctions, Optional Argument, Returning Value from Function. Control Flow Statements:ConditionalStatement,LoopStatement.Forms:Loading,ShowingandHidingForms,WorkingwithMultipleForms,ControllingoneFormwithinAnother,OverviewofC#,Structureof C# Program,C# in .Net.

UNIT-III

GUIProgramingwithWindowsFormwithProperties,MethodsandEvents:TextBoxControl, LabelControl,ButtonControl,Listbox,ComboBox,CheckedBox,PictureBox,RadioButton,Pannel, Scroll Bar, Timer Control, Adding Controls At Runtime, Common Dialog Control :File, Save, Print, Help. Designing Menus, MDI Forms, Overview of Dynamic Web Page,Asp.Net Controls, Applications, Web Servers, Web Form Controls, Server Controls, Client Controls Adding Controls to a Web Form, Form Validation Controls: Client Side Validation ServerSideValidation

UNIT-IV

ADO,.NetArchitecture,CreateConnection,AccessingDataUsingDataAdaptersandDatasets,UsingCommand&DataReader,DataBindControls,DisplayingDatainDataGrid.DataFormWizard,ProcessingSQL&AccessDatabaseUsingAdo.NetObjectModel,Connection Object, Command Object, Add, Delete, Move & Update Records to Dataset, Executing Queries

UNIT-V

XMLin.Net,XMLBasics,Attributes,FundamentalXmlClasses:Document,Textwriter,Textreader, XML Validations, XML in ADO, .Net, the XML data document. Web Services: State Management- View State, Session State, Application State, Web Service Description Language, Building & Consuming a Web Service. Web Application Deployment, Caching

ReferenceBooks:

1. StevenHolznerVB.NetProgramming-BlackBook-DreamtechPublications
2. EvangelosPetroutsosMasteringVB.Net-BPBPublications
3. MathewMacdonald-TheCompleteReference Asp.Net-TMH
4. ProfessionalASP.Net-WroxPublication
5. StephenWaltherActiveServerPages2.0(Unleashed)-Techmedia
6. Erica.SmithAsp3ProgrammingBible: IDG Books
7. C#Programming-WroxPublication
8. MattTelles-C#ProgrammingBlackBook-DreamtechPublication

DOT NET TECHNOLOGY LAB SUB-CODE:MCAL204 CREDIT: 02

Course Objective:

To enable the students to

- To learn about basic features of ASP.NET and its controls
- To create an ASP.NET application using standard .NET Controls
- To learn about connecting data sources using ADO.NET and managing them**.**

Course Outcomes:

At the end of the course the :

- Learners will be able to design applications using .NET
- Learners will be able to use .NET controls in applications
- Learners will be able to create database driven ASP.NET web applications and web services

Syllabus:

LIST OF EXPERIMENTS(about one per week)

1 Program to display the addition, subtraction, multiplication and division of two number using console application.

2 Program to display the first 10 natural numbers and their sum using console application.

3 Program to display the addition using the windows application.

4 Write a program to convert input string from lower to upper and upper to lower case.

5 Write a program to simple calculator using windows application.

6 Write a program working with Page using ASP.Net.

7 Write a program working with forms using ASP.NET.

8 Write a program to connectivity with Oracle database.

9 Write a program to access data source through ADO.NET.

10 Write a program to manage the session.

Programme Structure	MCA
Subject Code	MCAT 205
Course Name	Operating System
Course Credits	3 (T)=3
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**

This course will introduce the core concepts of operating systems, such as processes and threads, scheduling, synchronization, memory management, file systems, input and output device management and security. The goal of the programming assignments is to give students some exposure to operating system code.

2. **Prerequisite:** To understand the basic computer organization, operating system structures, processes and threads.

3. **Objective of the Syllabus:**

This course OPERATING SYSTEMS is an essential part of any Computer-Science education. The purpose of this course is to understand the mechanisms of the Operating Systems like Process Management, Process Synchronization, Memory Management, File System Implementation, Storage Structures used in OS and Protection Principles. How effectively the OS is utilizing the CPU resources with the help of these mechanisms.

Learn the fundamentals of Operating Systems, learn the mechanisms of OS to handle processes and threads and their communication. To learn the mechanisms involved in memory management in contemporary OS. To gain knowledge on distributed operating system concepts that includes architecture, Mutual exclusion algorithms, deadlock detection algorithms and agreement protocols.

4. **Course Outcomes:**

S.No.	CourseOutcomes(Cos)
CO1	Will be able to control access to a computer andthefiles that maybeshared.
CO2	Demonstrate the knowledge of the componentsof computer and their respective roles incomputing.
CO3	Abilitytorecognizeandresolveuserproblemswithstandardoperatingenvironments
CO4	Gain practical knowledge of how programminglanguages, operating systems, and architecturesinteractand how to use each effectively

5. Syllabus:

Unit I Introduction

Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS - Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine. Case study on UNIX and WINDOWS Operating System.

Unit II Processes

Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching Thread: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads, Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non pre-emptive, FCFS, SJF, RR; Multiprocessor scheduling: Real Time scheduling: RM and EDF.

Unit III Inter-Process Communication

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

Unit IV Deadlocks

Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker‟s algorithm, Deadlock detection and Recovery.

Memory Management: Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation – Fixed and variable partition– Internal and External fragmentation and Compaction Paging: Principle of operation – Page allocation – Hardware support for paging, Protection and sharing, Disadvantages of paging. Virtual Memory: Basics of Virtual Memory – Hardware and control structures – Locality of reference, Page fault , Working Set , Dirty page/Dirty bit – Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Second Chance (SC), Not recently used (NRU) and Least Recently used (LRU).

Unit V I/O Hardware

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: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed), Free-space management (bit vector, linked list, grouping), directory implementation (linear list, hash table), efficiency and performance. Disk Management: Disk structure, Disk scheduling - FCFS, SSTF, SCAN, C-SCAN, Disk reliability, Disk formatting, Boot-block, Bad blocks

BOOKS AND REFERENCES

Text Books

1. Operating System Principles, Abraham Silberchatz, Peter B.Galvin,Greg Gagne,8th Edition, Wiley Student Edition,
2. Operating System-Internals and Design Principles, W.Stallings, 6th Edition, Pearson,

Reference Books

1. Modern Operating System, Andrew s Tanenbaum, 3rd Edition, PHI
2. Operating System A concept-based Approach, 2nd Edition, D.M.Dhamdhere, TMH
3. Principle Of Operating Systems, B.LStuart, Cengage Learning, India Edition
4. Operating system, A.s.Godbole, 2nd Edition, TMH
5. An Introduction to Operating System, P.C.P.bhatt, PHI.

Programme Structure	MCA
Subject Code	MCAT 206
Course Name	Accounting and Financial Management
Course Credits	3 (T) =2
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**

   The objective of this course is to introduce problems of financial accounting such as measuring and reporting issues related to assets and liabilities and preparing the financial statements.

2. **Prerequisite:** To understand the basic concepts of credit , debit and use of accounting methods
3. **Objective of the Syllabus:**

   To familiarize students with the mechanics of preparation of financial statements, understanding corporate financial statements, their analysis and interpretation, role of IFRS in accounting discipline, and the concept of management quality analysis and wealth creation.
4. **Course Outcomes:**

S.No.	CourseOutcomes(Cos)
CO1	Will be able to learn basics of accounting
CO2	Demonstrate the knowledge of journal and ledgers
CO3	Ability to work with various accounting tools
CO4	Have a sound knowledge of methods of financial management.
CO5	

5. **Syllabus:**

UNIT-I

Basic of Accounting, Accounting Mechanics- Double Entry System, Classification, RulesforDebitandCreditConcepts&Conventions,IndianAccounting Standards.

UNIT-II

Journal: Meaning of Journal, Advantages, Subdivision. Ledger : Meaning, subdivision,Mechanics of Posting, balancing of Ledger accounts .Trial Balance: Objectives, Defects oftrialbalance, Errorsdisclosedbytrialbalance,preparationandlocatingerrors.

UNIT–III

Cash Book and Subsidiary books of Accounting: Kinds of cashbook, Purchase daybook,Salesdaybook,Billsreceivablebook,Billspayablebook.FinanceAccounts:Tradingaccount,, Profit & Loss account, Adjustments, Balance Sheet, Forms of balance Sheet,Assetsandtheir classification, liabilities andtheirclassification,usesandlimitations.

UNIT-IV

Capital & Revenue Expenditure & Receipts: Rules for determining capital expenditure,Deferred Revenue expenditure, Capital & Revenue receipts, Capital & Revenue Profits,Capital & Revenue Loss. Nature of Financial Management: Scope of financial functions,finance functions andjobof finance manager, organization of finance function.
Suggested Reading:

1. Management Accounting –ManmohanSinghandGoel
2. Financial management-PandeyI.M.
3. Hanif & Mukherjee-Modern Accountancy, TMH,NewDelhi.
4. Maheshwari&Maheshwari-AnIntroductiontoAccountancy,VikasPublishingHousePvt.Ltd.,NewDelhi.

SEMESTER III

Programme Structure	MCA
Subject Code	MCAT301
Course Name	Design And Analysis of Algorithms
Course Credits	3 (T) +1 (P)=4
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**
   This course is intended to make students familiar with some basic graph algorithms and their efficiency analysis . Also this course will provide a detailed introduction to different algorithm design paradigms with illustrative problems. The course is looking at designing algorithms for the classical network flow problem.

2. **Prerequisite:** To understand the basic programming and data structures.

3. **Objective of the Syllabus:**
   i. Analyze the asymptotic performance of algorithms.
   ii. Write rigorous correctness proofs for algorithms.
   iii. Demonstrate a familiarity with major algorithms and data structures.
   iv. Apply important algorithmic design paradigms and methods of analysis.
   v. Synthesizeefficientalgorithmsincommonengineeringdesignsituations.

4. **Course Outcomes:**

S.No.	CourseOutcomes(Cos)
CO1	To develop the ability to analyze the running time and prove the correctness of basic algorithms.
CO2	To be able to design efficient algorithms for moderately difficult computational problems, using various algorithm design techniques taught in the course.
CO3	To be able to prove the hardness of NP-Hard problems using simple reductions.
CO4	To be able to do performance analysis of simple approximation algorithms.
CO5	

5. Syllabus:

UNITI

INTRODUCTION

Characteristicsofalgorithm.Analysisofalgorithm:Asymptoticanalysisofcomplexitybounds – best, average and worst-case behavior; Performance measurements of Algorithm, Time and space trade- offs, Analysis of recursive algorithms through recurrence relations: Substitution method ,Recursion tree method and Masters theorem.

UNITII

FUNDAMENTAL ALGORITHMIC STRATEGIES

Brute-Force, Greedy, Dynamic Programming, Branch-and-Bound and Backtracking methodologies for the design of algorithms; Illustrations of these techniques for Problem-Solving, Bin Packing, KnapSackTSP. Heuristics–characteristic sand their application domains.

UNITIII

GRAPHANDTREEALGORITHMS ***Traversalalgorithms: DepthFirstSearch (DFS) and Breadth First Search(BFS); Shortest path algorithms, Transitive closure, Minimum Spanning Tree, Topological sorting, Network Flow Algorithm.***

UNITIV

TRACTABLEANDINTRACTABLEPROBLEMS

Computability of Algorithms, Computability classes – P, NP, NP-complete and NP-hard.Cook"stheorem, StandardNP-complete problemsand Reductiontechniques.

UNITV

ADVANCEDTOPICS

Approximation algorithms, Randomized algorithms,ClassofproblemsbeyondNP-PSPACE

TEXTBOOKS:

1. IntroductiontoAlgorithms,4THEdition,ThomasHCormen,CharlesELieserson,RonaldLRivestandCliffordStein,MITPress/McGraw-Hill.
2. FundamentalsofAlgorithms –E.Horowitz etal.

REFERENCES:

1. AlgorithmDesign,1STEdition,JonKleinbergandÉvaTardos,Pearson.
2. AlgorithmDesign:Foundations,Analysis,andInternetExamples,Second Edition,MichaelTGoodrich andRoberto Tamassia,Wiley.
3. Algorithms--ACreativeApproach,3rd Edition,UdiManber,Addison-Wesley,Reading,MA.

DESIGN & ANALYSIS OF ALGORITHMS LAB SUB-CODE: MCAL 301 CREDIT: 02

Course Objective:

Students will be able to do the following: Analyze the asymptotic performance of algorithms. Write rigorous correctness proofs for algorithms. Demonstrate a familiarity with major algorithms and data structures.

Course Outcomes:

At the end of the course the students are able to:

- To develop the ability to analyze the running time and prove the correctness of basic algorithms.
- To be able to design efficient algorithms for moderately difficult computational problems, using various algorithm design techniques taught in the course.
- To be able to prove the hardness of NP-Hard problems using simple reductions.
- To be able to do performance analysis of simple approximation algorithms.

Syllabus:

1. Implementation of Binary search algorithm using Divide & Conquer method.
2. Implementation of Quick Sort algorithm using Divide & Conquer method.
3. Program to merge two sorted arrays.
4. Implementation of Merge Sort algorithm using Divide & Conquer method.
5. Implementation of Strassen's Matrix multiplication.
6. Program to implement knapsack problem using greedy method.
7. Program to implement job sequencing with deadlines using greedy method.
8. Implementation of travelling salesman problem.
9. Program to implement 8-queens problem using backtrack method.
10. Implement All-Pairs Shortest Paths Problem using Floyd's algorithm.

Programme Structure	MCA
Subject Code	MCAT302
Course Name	Software Engineering
Course Credits	3 (T) =3
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**

   This course introduces students to the different software development lifecycle (SDLC) phases used in developing, delivering, and maintaining software products. Students will also acquire basic software development skills and understand common terminology used in the software engineering profession. Students will also learn and practice using traditional coding standards/guidelines.

2. **Prerequisite:** To understand the basic software development , programming ,databases etc.

3. **Objective of the Syllabus:**
   - List and describe the fundamental phases of the Software Development Lifecycle (SDLC)
   - Define and describe fundamental software engineering terminology and coding practices
   - Explore/explain relationships between software engineering and other engineering disciplines (Systems Engineering, Electrical and Computer Engineering, Industrial Engineering)
4. **Course Outcomes:**

S.No.	CourseOutcomes(Cos)
CO1	Modify/build a software program that introduces students to software development tools / environments
CO2	Troubleshoot and debug changes made to an existing software program
CO3	Ability to work out best possible model for development of a software
CO4	Gain practical knowledge of how programming languages, operating systems, and architectures interact and how to use each effectively
CO5	

5. **Syllabus:**

UNITI

IntroductionofSoftwareEngineering&SoftwareLifeCycleModels/WEngineeringDisciplin e-
EvolutionandImpact,EmergenceofS/WEngineering,WaterfallModel,PrototypingModel, EvolutionarybModel Spiral models and their comparisons.

UNITII

Software Project Management & Requirements Analysis and Specification, Project ManagerResponsibilities,ProjectPlanning,ProjectSizeestimationMetrics,Projectestimat ionTechniques,COCOMO,StaffingLevelEstimation,Scheduling,Organization&TeamStruc tures,Staffing,RiskManagement,S/WConfigurationManagement,requirementGatherin g and Analysis, Software Requirement Specification, Formal System Development Techniques

UNITIII

Software Design & Function-Oriented S/W Design, Overview of Software Design, CohesionandCoupling,S/WDesignApproaches,Object-Orientedvs.Function-Oriented Design, SA/SD Methodology, Structured Analysis, Data Flow Diagrams(DFDs), Structured Design, Detailed Design, Design Preview

UNITIV

Object Modeling Using UML &Object-Oriented Software Development , UML , UML Diagrams, Use Case Model, Class Diagrams, Design Patterns, Object-Oriented analysis and Design Process, OOD Goodness Criteria

UNITV

UserInterfaceDesign&CodingandTesting,BasicConcepts,Types,ComponentsBasedGUI, Development,UserInterfaceDesignMethodology,Coding,CodeReview,UnitTesting,Blac kBox, Testing, White-Box Testing, Debugging, Program Analysis Tools, Integration Testing,SystemTesting,SoftwareReliabilityandQualityManagement&SoftwareMainten ance,S/WReliability,Statistical,Testing,S/WQuality,ManagementSystem,ISO9000,SEIC MM,S/WReverseEngineering, S/W, Maintenance Process Models,

Textbooks:

1. R.S.Pressman,SoftwareEngineering:APractitionersApproach,McGrawHill.
2. Rajib Mall , Fundamentals of Software Engineering ,PHIPublication.
3. K.K.AggarwalandYogeshSingh,SoftwareEngineering,NewAgeInt ernationalPublishers.
4. Pankaj Jalote , Software Engineering,Wiley

Programme Structure	MCA
Subject Code	MCAT303
Course Name	Advanced Java
Course Credits	3 (T) +1 (P)=4
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**
   Java is platform independent language and Object oriented Programming language. Using advanced Java programming language we can learn how to design dynamic web applications using Java Server Pages and Java Servlet and how to connect to data base drivers. Advanced java course consist JDBC, HTML, Servlet, JSP and JSTL. Using JDBC concept you can learn database concepts in depth and perform all CRUD operations easily. Using HTML you can develop static web pages. Using Servlet and JSP you can develop dynamic web pages.

2. **Prerequisite:** To understand Java,AWT,Swing and networking.

3. **Objective of the Syllabus:**
   This course is a part of Java programming language. It is an advanced technology or advance version of Java specially designed to develop web-based, network-centric or enterprise applications. It includes the concepts like Servlet, JSP, JDBC, RMI, Socket programming, etc.

4. **Course Outcomes:**

S.No.	CourseOutcomes(Cos)
CO1	To become familiar with the advanced features of Java Language.
CO2	To develop Web Applications using Servlets / JSP and deploy in popular servers like Tomcat.
CO3	To understand Java Servlets and their life cycle and to understand Java Web application directory structure
CO4	To understand Java Web application directory structureϖ

5. Syllabus:

UNIT-I

Servlets:Servlet Structure, Servlet packaging , HTML building utilities ,Lifecycle, SingleThreadmodelinterface,HandlingClientRequest:FormData,HandlingClientRequest:HTTP RequestHeaders.GeneratingserverResponse:HTTPStatuscodes,GeneratingserverResponse:HTTP Response Headers, Handling Cookies ,Session Tracking.

UNIT-II

JSP : Overview of JSP Technology, Need of JSP, Benefits of JSP, Advantages of JSP,Basicsyntax,InvokingjavacodewithJSPscriptingelements,creatingTemplateText,Invokingjavacode from JSP, Limiting java code in J S P , using jsp expressions ,comparing servlets and jsp,writingscriptlets.ForexampleUsingScriptletstomakepartsofjspconditional,usingdeclarations,declarationexample.ControllingtheStructureofgeneratedservlets:theJSPpagedirective,importattribute,sessionattribute,isElignoreattribute,bufferandautoflushattributes,infoattribute, errorPageandiserrorPageattributes,isThreadsafeAttribute,extendsattribute,languageattribute,IncludingfilesandappletsinjspPages,usingjavabeanscomponentsin JSP documents

UNIT-III

Java Beans & Annotations: Creating Packages, Interfaces, JAR files and Annotations. The core java API package , New java .Lang Sub package ,Built-inAnnotations.Workingwith JavaBeans. Introspection, Customizers, creating java bean, manifest file, Bean Jar file, new bean ,adding controls, Bean properties, Simple properties, Design Pattern events, creating bound properties, Bean Methods ,Bean info class ,Persistence ,Java BeansAPI.

UNIT-IV

JDBC:TalkingtoDatabase,ImmediateSolutions,EssentialJDBCprogram,usingpreparedStatementObject,InteractiveSQLtool.JDBCinActionResultsets,Batchupdates,Mapping,Basic JDBC datatypes, Advanced JDBC datatypes,immediate solutions.

UNIT-V

IntroductiontoEJB:TheProblemdomain,Breakupresponsibilities,CodeSmartnothard,theEnterprisejavabeanspecification.ComponentsTypes.ServerSideComponentTypes,SessionBeans, Message Driven Beans, Entity Beans, The Java Persistence Model. Container services.DependencyInjection,Concurrency,Instancepoolingncaching,Transactions,security,Timers,Namingandobjectstores,Interoperability,LifeCycleCallbacks,Interceptors,platformintegration.DevelopingyourfirstEJB.preparation,Definitions,namingconventions,conventionfortheExamples,codingtheEJB,thecontract,thebeanImplementationclass,out

ofContainerTesting,IntegrationTesting.

ReferenceBooks:

1. E.Balagurusamy,"ProgrammingwithJava,aPrimer",TMH,ISBN-13:978-0-07-061713-
2. Patrick Naughton and Herbert Schildt,"Java:theCompleteReference", TMHPublication.
3. Yashavant kanetkar,"Let us Java",BPBPublications.
5. Cay Horstmann, "Big Java", Wiley Publication
6. Peter Norton, "Java Programming", Techmedia Publications.
7. Joseph Weber,"Using Java 1.2", PHI

ADVANCED JAVA LAB SUB-CODE:MCAL303 CREDIT: 02

Course Objective:

- To develop Server side components in a Java Web application
- To understand Java server Pages (JSP),Servlets and EJB technology

Course Outcomes:

At the end of the course the students are able to:

- To develop reusable components using JavaBeans.
- To develop JSP pages using which use JavaBeans
- To develop JSP Custom tags and use them in JSP pages
- Writing Event Listeners in Java Web application
- Handling File uploads in Java Web application
- To discover how to write Java applications this can communicate with Relational Databasesϖ

Syllabus:

LISTOFEXPERIMENTS

1 Experiment : JDBC Programs using Statement

1.1 A program to test the connection with the database

1.2 A program to create a table

1.3 A program to insert record in a table

1.4 A program to update record in a table

1.5 A program to delete record from a table

2 Experiment : Servlet Programming

2.1 Servlet Execution on tomcat

2.2 A servlet program to print hello world

2.3 A servlet program to display request details

2.4 A servlet program to handle user form

3 Experiment : JSP Programming

Programme Structure	MCA
Subject Code	MCAT304
Course Name	Data Mining
Course Credits	3 (T)=3
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**

   This course will introduce the concepts of data ware house and data mining, which gives a complete description about the principles, used, architectures, applications, design and implementation of data mining and data ware housing concepts.

2. **Prerequisite:** To understand the basic DBMS and SQL queries etc

3. **Objective of the Syllabus:**

   To introduce students to the basic concepts and techniques of Data Mining ,to develop skills of using recent data mining software for solving practical problems And to gain experience of doing independent study and research.

4. **Course Outcomes:**

S.No.	CourseOutcomes(Cos)
CO1	Be familiar with mathematical foundations of data mining tools.
CO2	Understand and implement classical models and algorithms in data warehouses and data mining
CO3	Characterize the kinds of patterns that can be discovered by association rule mining
CO4	Develop skill in selecting the appropriate data mining algorithm for solving practical problems.
CO5	

5. **Syllabus:**

UNIT-I

Introduction to Data Mining : What is data mining?; Related technologies - MachineLearning,DBMS,OLAP,Statistics;DataMiningGoals;StagesoftheDataMiningProcess;Data Mining Techniques ; Knowledge Representation Methods; Applications ;Example : weather data.

UNIT-II

Data Warehouse and OLAP: Data Warehouse and DBMS; Multidimensional data model OLAP operations; Example: loan dataset. Data preprocessing: Data cleaning; Data transformation; Data reduction; Discretization and generating concept hierarchies ; InstallingWeka3 Data Mining System; Experiment swith Weka-filters, discretization.

UNIT-III

Data Mining Primitives, Primitives, Languages and System Architectures : Data MiningPrimitives, Data Mining query language, Designing GUI on a Data Mining query language , Architectures of Data Mining System.

UNIT-IV

MiningAssociationrulesinlargedatabase:Associationrulesmining,Miningsingle-dimensional Boolean Association rules from transaction database, mining multilevel Association rules from transaction database, Mining multidimensional Association rules from relational databases and Data warehouses , Association mining to correlation analysis ,Constraint based association mining.

UNIT-V

ClassificationandPrediction:Whatisclassificationandprediction,Issuesregardingclassificationandprediction,ClassificationbydecisiontreeInduction,BayesianClassification,Classification by Back propagation, Classification based on concepts from association rulemining,Prediction,Classificationaccuracy.lusterAnalysis:Whatisclusteranalysis,Typesofdata in cluster analysis, Acategorization of major clustering methods, Partitioning methods,Hierarchical Methods, Density based methods, Grid based methods, Model based clusteringmethods.

SuggestedReadings/References:-

1. DataMiningConceptsandTechniquesbyJiaweiHan,MichelineKamber,Elsevier.
2. DataMining.Atutorial-basedPrimerbyRoiger,MichaelW.Geatzand PearsonEducation.
3. DataMiningIntroductory&advancedtopicbyMargaretH.Durham,PearsonEducation

Programme Structure	MCA
Subject Code	MCAT305
Course Name	Python Programming
Course Credits	3 (T) +1 (P)=4
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**

Python is an interpreted , object-oriented, high-level programming language with dynamic semantics. Its high level built in data structures, combined with dynamic typing and dynamic binding , make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed The goal of python programming is to give students some exposure to intelligent coding.

2. **Prerequisite:** To understand the basic computer organization, operating system structures, processes and threads.

3. **Objective of the Syllabus:**

- To know the basics of algorithmic problem solving
- To read and write simple Python programs.
- To develop Python programs with conditionals and loops.
- To define Python functions and call them.
- To use Python data structures–-lists , tuples , dictionaries.
- Todoinput/outputwithfilesinPython.

4. **Course Outcomes:**

S.No.	Course Outcomes(Cos)
CO1	Develop algorithmic solutions to simple computational problems
CO2	Read , write , execute by hand simple Python programs.
CO3	Structure simple Python programs for solving problems.
CO4	Decompose a Python program into functions.

CO5	Represent compound data using Python lists, tuples, dictionaries.

5. Syllabus:

UNIT I

ALGORITHMICPROBLEMSOLVING

Algorithms, building blocks of algorithms (statements, state, control flow, functions),notation (pseudo code, flow chart, programming language), algorithmic problem solving, simple strategies for developing algorithms (iteration, recursion). Illustrative problems: find minimum in a list, insert a card in a list of sorted cards, guess an integer number in arrange, Towers of Hanoi.

UNIT II

DATA,EXPRESSIONS,STATEMENTS

Python interpreter and interactive mode; values and types: int, float, boolean, string, andlist; variables, expressions, statements, tupleassignment, precedence of operators, comments; modules and functions, function definition and use, flow of execution, parameters and arguments; Illustrative programs: exchange the values of two variables, circulate the values of variables, distance between two points.

UNIT III

CONTROLFLOW,FUNCTIONS

Conditionals:Booleanvaluesandoperators,conditional(if),alternative(if-else), chained conditional (if-elif-else); Iteration: state, while, for, break, continue, pass; Fruitful functions: return values, parameters, local and global scope, function composition,recursion;Strings:stringslices,immutability,stringfunctionsandmethos, string module; Lists as arrays. Illustrative programs: square root, gcd, exponentiation, sum an array of numbers, linear search, binary search.+

UNIT IV

LISTS,TUPLES,DICTIONARIES

Lists: list operations, list slices, list methods, list loop, mutability, aliasing, cloning lists, list parameters; Tuples: tuple assignment, tuple as return value; Dictionaries: operations and methods; advanced list processing-list comprehension; Illustrative programs: selection sort, insertion sort, merge sort, histogram.

UNIT V

FILES,MODULES,PACKAGES

Files and exception: text files, reading and writing files, format operator;

command line arguments, errors and exceptions, handling exceptions, modules, packages; Illustrative programs: word count, copy file.

TEXTBOOKS:

1. Allen B. Downey, ``Think Python: How to Think Like a Computer Scientist„„„2ndedition,UpdatedforPython3,Shroff/O„ReillyPublishers,2016(http://greenteapress.com/wp/think-python/)
2. Guido van Rossum and Fred L. Drake Jr, —An Introduction to Python – RevisedandupdatedforPython 3.2, NetworkTheoryLtd.,2011.

REFERENCEBOOKS:

1. JohnVGuttag,—IntroductiontoComputationandProgrammingUsin gPython„„„Revisedandexpanded Edition,MIT Press , 2013
2. RobertSedgewick,KevinWayne,RobertDondero,—Introductionto ProgramminginPython:An Inter-disciplinaryApproach,PearsonIndiaEducationServicesPvt.Ltd., 2016.
3. TimothyA.Budd,—ExploringPython‖,Mc-Graw Hill Education(India)PrivateLtd.,,2015.
4. KennethA.Lambert,—FundamentalsofPython:FirstPrograms‖,CENGAGEL earning,2012.
5. CharlesDierbach,—IntroductiontoComputerScienceusingPython:ACo mputationalProblem-SolvingFocus,WileyIndiaEdition, 2013.
6. PaulGries,JenniferCampbellandJasonMontojo,—PracticalProgramming:

PYTHON PROGRAMMING LAB SUB-CODE:MCAL305 CREDIT: 02

Course Objective:

Become technology-oriented with the knowledge and ability to develop creative solutions, and better understand the effects of future developments of computer systems and technology on people and society as a whole. Acquire some development experience within a specific field of Computer Science, through project work. Gain ability to apply knowledge of Python to the real-world issues.

Course Outcomes:

At the end of the course the students are able to:

- an ability to handle Python programs
- an ability to use functions and classes in Python
- an understanding of the concepts of OOPs including inheritance and polymorphism
- an ability to overload operators in Python
- an understanding of function overloading

Syllabus:(One or two Programs per week)

1 Write a program to demonstrate different number data types in python(script.py)

2 Write a program to perform different arithmetic operations on numbers in python

3 Write a program to create, concatenate and print a string and accessingsub-string from given string

4 Write a python script to print the current date in the fallowing format"SunMay 29 02:26:23 IST 2017"

5 Write a program to create, append and remove lists in python

6 Write a program to demonstrate working with tuples in python

7 Write a program to demonstrate working with dictionaries in python

8 Program to find the largest number among the three input numbers

9 Program to convert temperature in Celsius to Fahrenheit

10 Write a python program to construct the following pattern, using a nested for loop

11 Write a python script that prints prime numbers less than 20

12 Write a python program to find the factorial of a number using recursion

13 write a program that accepts the lengths of three sides of a triangle asinput the program output should indicate whether or not the triangle, isright triangle(recall from the Pythagorean theorem that in a right triangle,the square of one side equals the sum of the squares of the other two sides)

14 Write a python class to implement pow(x, n)

15 Write a python class to reverse a string word by word.

Programme Structure	MCA
Subject Code	MCAT306
Course Name	Artificial Intelligence
Course Credits	4 (T) =4
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**
   Artificial Intelligence is a method of making a computer, a computer-controlled robot, or a software think intelligently like the human mind. AI is accomplished by studying the patterns of the human brain and by analyzing the cognitive process. The outcome of these studies develops intelligent software and systems.

2. **Prerequisite:** To understand the basic Data Structures and Algorithms Analysis of Algorithm
3. **Objective of the Syllabus:**
   - Study the concepts of Artificial Intelligence.
   - Learn the methods of solving problems using Artificial Intelligence.
   - Learn the knowledge representation techniques, reasoning techniques and planning
   - Introduce the concepts of Expert Systems and machine learning.

4. **Course Outcomes:**

S.No.	CourseOutcomes(Cos)
CO1	Understand the informed and uninformed problem types and apply search strategies to solve them.
CO2	Apply difficult real life problems in a state space representation so as to solve them using AI techniques like searching and game playing.
CO3	Design and evaluate intelligent expert models for perception and prediction from intelligent environment.
CO4	Formulate valid solutions for problems involving uncertain inputs or outcomes by using decision making techniques
CO5	

5. **Syllabus:**

UNIT I

Introduction,What is Artificial Intelligence ,ITechnique , Level of the Model , Problem Spaces ,and Search, Defining the Problem as a State Space Search, Production Systems, Problem,Characteristics, Production System Characteristics, Issues in the

Design of Search Programs, Heuristic Search Techniques: Generate-and-Test, Hill Climbing, Best-first Search, Problem Reduction, Constraint Satisfaction, Means-end Analysis, knowledge, Means-ends Analysis, Knowledge

UNIT II

Using Predicate Logic, Representing Simple Facts in Logic, Representing , Instance and ISA Relationships, Computable Functions and Predicates, Resolution, Natural Deduction, Introduction to Non monotonic Reasoning , Logics for Non monotonic Reasoning , Implementation Issues, Augmenting a Problem-solver, Depth-first Search, Breadth-first Search, Semantic Nets, Frames, Conceptual Dependency Scripts, Weak and Strong Slot-and-Filler Structures: Semantic

UNIT III

GamePlaying,TheMinimaxSearchProcedure,AddingAlpha-betaCutoffs,IterativeDeepening,TheBlocks World, Components of a Planning System , Components of a Planning System, Goal Stack Planning, Nonlinear Planning Using Constraint Posting, Hierarchical Planning Other Planning Techniques. What is Understanding, What Makes Understanding Hard? Understanding as Constraint Satisfaction

UNIT IV

Learning, Rote Learning, Learning by Taking Advice, Learning by Taking Advice, LearninginProblem-solving,Learning,LearningfromExamples:Induction,Explanation-basedLearning, Discovery, Analogy, Formal Learning Theory, Neural Net Learning and Genetic Learning

UNIT V

Expert Systems, Representing and Using Domain Knowledge, Expert System Shells Knowledge Acquisition

REFERENCES:

1. Stuart Russell, Peter Norvig, “Artificial Intelligence – A Modern Approach”, PearsonEducation
2. ElaineRichandKevin Knight,“ArtificialIntelligence”,McGraw-Hill
3. E Charniak and D McDermott, “Introduction to Artificial Intelligence”, PearsonEducation
4. DanW.Patterson,“ArtificialIntelligenceandExpertSystems”,PrenticeHallofIndia,

Programme Structure	MCA
Subject Code	MCAT307
Course Name	Parallel Computing
Course Credits	4 (T)=4
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**
   This course will introduce the core concepts of parallel computing.
   Parallel computing is a type of computing architecture in which several processors simultaneously execute multiple, smaller calculations broken down from an overall larger, complex problem.
2. **Prerequisite:** To understand the distributed computing concepts and AI based concepts

3. **Objective of the Syllabus:**

   The course is structured so that the participants understand challenges in efficient execution of large-scale parallel applications. The assignments will be designed to strengthen understanding of parallel programming. The course will also involve a research-based project component.

4. **Course Outcomes:**

S.No.	CourseOutcomes(Cos)
CO1	Solve the Problems in Parallel
CO2	Have knowledge on Different Structures of Parallel Computers
CO3	Understand the Performance Evaluation of Parallel Computers
CO4	Understand Parallel programming

5. **Syllabus:**

UNIT I

Scalability And Clustering: Evolution of Computer Architecture–Dimensions of Scalability

– Parallel Computer Models – Basic Concepts Of Clustering – Scalable Design Principles – Parallel Programming Overview–Processes, Tasks and Threads –Parallelism Issues– Interaction/Communication Issues–Semantic Issues In Parallel Programs.

UNIT II

Enabling Technologies : System Development Trends – Principles of Processor Design – Microprocessor,ArchitectureFamilies–HierarchicalMemoryTechnology–Cache Coherence Protocols– Shared Memory Consistency–Distributed Cache Memory Architecture– Latency Tolerance Techniques –Multithreaded Latency Hiding.

UNIT III

System Interconnects: Basics of Interconnection Networks–Network Topologies and Properties–Buses, Crossbar and Multistage Switches, Software Multithreading–Synchronization Mechanisms.

UNIT IV

Parallel Programming: Paradigms And Programmability – Parallel Programming Models – Shared Memory Programming.

UNIT V

Message Passing Programming:

Message Passing Paradigm –Message Passing Interface–Parallel Virtual Machine.

TEXTBOOK

1. Kai Hwang and Zhi.WeiXu, “Scalable Parallel Computing”, Tata McGraw-Hill,NewDelhi, 2003.

REFERENCES

1. David E. Culler &Jaswinder Pal Singh, “Parallel Computing Architecture: AHardware/SoftwareApproach”,Morgan KaufmanPublishers, 1999.
2. Michael J. Quinn, “Parallel Programming in C with MPI &OpenMP”, Tata McGraw-Hill,New Delhi, 2003.
3. Kai Hwang, “Advanced Computer Architecture” Tata McGraw-Hill, New Delhi,2003.

Programme Structure	MCA
Subject Code	MCAT308
Course Name	Cyber Security
Course Credits	4 (T) =4
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**

   Cyber security is the practice of protecting systems, networks, and programs from digital attacks. These cyber attacks are usually aimed at accessing, changing, or destroying sensitive information; extorting money from users via ransomware; or interrupting normal business processes.

2. **Prerequisite:** To understand the basic computer organization, operating system and networking.

3. **Objective of the Syllabus:**
   - Exhibit knowledge to secure corrupted systems, protect personal data, and secure computer networks in an Organization.
   - Practice with an expertise in academics to design and implement security solutions.
   - Understand key terms and concepts in Cryptography, Governance and Compliance. Develop cyber security strategies and policies
   - Understand principles of web security and to guarantee a secure network by monitoring and analyzing the nature of attacks through cyber/computer forensics software/tools.

4. **Course Outcomes:**

S.No.	CourseOutcomes(Cos)
CO1	Analyze and evaluate the cyber security needs of an organization.
CO2	Determine and analyze software vulnerabilities and security solutions to reduce the risk of exploitation.
CO3	Measure the performance and troubleshoot cyber security systems.
CO4	Implement cyber security solutions and use of cyber security

5. Syllabus:

UNIT I

Introduction to CyberSecurity – Importance and challenges in Cyber Security-Cyberspace – Cyber threats - Cyber warfare - CIA Triad - Cyber Terrorism - Cyber Security of Critical Infrastructure-Cyber security-Organizational Implications.

UNIT II

Hackers and Cyber Crimes

Types of Hackers - Hackers and Crackers - Cyber-Attacks and Vulnerabilities - Malware threats -Sniffing - Gaining Access - Escalating Privileges - Executing Applications - Hiding Files–Covering Tracks-Worms -Trojans-Viruses– Backdoors

UNIT III

Ethical Hacking and Social Engineering

EthicalHackingConceptsandScopes-ThreatsandAttackVectors-InformationAssurance– Threat Modeling - Enterprise Information Security Architecture - Vulnerability Assessment and Penetration, Testing - Types of Social Engineering - Insider Attack - Preventing Insider Threats– Social Engineering Targets and Defense Strategies.

UNIT IV

Cyber Forensics and Auditing

IntroductiontoCyberForensics-ComputerEquipmentandassociatedstoragemedia-Role of forensics Investigator-Forensics Investigation Process-Collecting Network based Evidence Writing Computer , Forensics Reports-Auditing-Plan an audit against a set of audit criteria Information Security, Management System Management. Introduction to ISO27001:2013.

UNIT V

Cyber Ethics and Laws

Introduction to Cyber Laws - E-Commerce and E-Governance - Certifying Authority and Controller -Offences under IT Act- Computer Offences and its penalty under IT Act 2000 Intellectual Property Rights in Cyberspace.

BooksforReferences:

Donaldson, S., Siegel, S., Williams, C.K., Aslam, A., “Enterprise Cyber security -How toBuild a Successful Cyber defense Program against Advanced Threats”, Apress, 1st Edition,2015.

NinaGodbole,SumitBelapure,“CyberSecurity”,Willey,2011.

Roger Grimes,“Hackingthe Hacker”,Wiley,ist Edition, 2017.

CyberLaw ByBareAct,GovtofIndia, ItAct2000

Programme Structure	MCA
Subject Code	MCAT309
Course Name	Cloud Computing
Course Credits	4 (T) =4
Total Course Credit	92

Abbreviations: T-Theory, P-Practical

1. **Course Overview:**
   Cloud computing is the on-demand access of computing resources—physical servers or virtual servers, data storage, networking capabilities, application development tools, software, AI-powered analytic tools and more—over the internet with pay-per-use pricing.

2. **Prerequisite:** To understand operating systems , networking ,databases and servers.

3. **Objective of the Syllabus:**
   - To provide students with the fundamentals and essentials of Cloud Computing.
   - To provide students a sound foundation of the Cloud Computing so that they are able to start using and adopting Cloud Computing services and tools in their real life scenarios.
   - To enable students exploring some important cloud computing driven commercial systems and applications.
   - To expose the students to frontier areas of Cloud Computing and information systems, while providing sufficient foundations to enable further study and research.

4. **Course Outcomes:**

S.No.	CourseOutcomes(Cos)
CO1	Explain the core concepts of the cloud computing paradigm: how and why this paradigm shift came about, the characteristics, advantages and challenges brought about by the various models and services in cloud computing.
CO2	Apply the fundamental concepts in datacenters to understand the tradeoffs in power, efficiency and cost.
CO3	Identify resource management fundamentals, i.e. resource abstraction, sharing and sandboxing and outline their role in managing infrastructure in cloud computing.
CO4	Analyze various cloud programming models and apply them to solve problems on the cloud.

5. **Syllabus:**

UNIT-I

Introduction: Objective, scope and outcome of the course. Introduction Cloud Computing: Nutshell of cloud computing, Enabling Technology, Historical development, Vision, feature Characteristics and components of Cloud Computing. Challenges, Risks and Approaches of Migration into Cloud. Ethical Issue in Cloud Computing, Evaluating the Cloud's Business Impact and economics, Future of the cloud. Networking Support for Cloud Computing . Ubiquitous Cloud and the Internet of Things

UNIT-II

Cloud Computing Architecture : Cloud Reference Model ,Layer and Types of Clouds, Services models, Data centre Design and interconnection Network, Architectural design of Compute and Storage Clouds. Cloud Programming and Software: Fractures of cloud programming , Parallel and distributed programming paradigms-Map Reduce, Hadoop, High level Language for Cloud. Programming of Google App engine.

UNIT-III

Virtualization Technology :Definition , Understanding and Benefits of Virtualization. Implementation Level of Virtualization , Virtualization Structure Tools and Mechanisms ,Hypervisor VMware, KVM, Xen. Virtualization: of CPU, Memory, I/O Devices, Virtual Cluster and Resources Management, Virtualization of Server, Desktop, Network, and Virtualization of data-centre.

UNIT-IV

Securingthe Cloud: Cloud Information security fundamentals ,Cloud security services, Design principles, Policy Implementation, Cloud Computing Security Challenges, Cloud Computing Security Architecture . Legal issues in cloud Computing. Data Security in Cloud: Business Continuity and Disaster Recovery , Risk Mitigation , Understanding and Identification of Threats in Cloud ,SLA-Service Level Agreements, Trust Management

UNIT-V

Cloud Platforms in Industry: Amazon web services , Google AppEngine, Microsoft AzureDesign, Aneka: Cloud Application Platform -Integration of Private and Public Clouds

Cloudapplications:Proteinstructureprediction,DataAnalysis,SatelliteImageProcessing,CRM

Text:

- Cloud computing a practical approach - Anthony T.Velte , Toby J. Velte RobertElsenpeter,
  TATAMcGraw- Hill,NewDelhi –2010
- Cloud Computing: Web-Based Applications That Change the Way You Work andCollaborate
  Online-Michael Miller-Que2008

References:

- Cloud computing for dummies- Judith Hurwitz , Robin Bloor , Marcia Kaufman ,FernHalper,WileyPublishing,Inc, 2010
- Cloud Computing (Principles and Paradigms), Edited by RajkumarBuyya, JamesBroberg,
  AndrzejGoscinski,JohnWiley&Sons,Inc. 2011

SEMESTER-IV

MCAL401 **L-T-P(20-0-0)** **Credit-20**

Dissertation (Major Project)

MCAL402 **L-T-P(4-0-0)** **Credit-4**

Seminar