

Course Name: Data Structures

Course Code: MCA121

Objectives:

- To develop proficiency in the specification, representation, and implementation of Data Types and Data Structures.
- To be able to carry out the Analysis of various Algorithms for mainly Time and Space Complexity.
- To get a good understanding of applications of Data Structures.
- To develop a base for advanced computer science study.

Prerequisites:

Any programming language like C, C++

Contents:

1 Introduction:

- Data types , ADT, data structure: Definition & classification
- Analysis of algorithms (recursive and non-recursive) with emphasis on best case, average case and worst case [5%]

2 Linear Data structures with applications:

- Array data structure: storage, mapping, applications (sparse matrix, polynomial representation, strings)
- List: Introduction, implementation using array & linked list (singly, doubly, circular, multi-list), Applications: Polynomial representation, Sparse matrix
- Stack: Introduction, implementation using array & linked list, Applications: Function call, Recursion, balancing of parenthesis, Polish Notation: infix to postfix conversion and evaluation of postfix expression [30%]
- Queue: Introduction (queue, circular queue, deque, priority queue), implementation using array & linked list, Applications: Job Scheduling

3 Non Linear data structures:

- Tree: Introduction and representation, Forest, Tree traversal, Binary Tree (representation using array and links): Binary tree traversal (recursive & non-recursive implementation), Expression tree [35%]
- Graph: Introduction, representations, Traversal(BFS, DFS), Applications: Shortest path (Single source-all destinations), Minimal spanning tree (Prim's algorithm, Kruskal's algorithm)

4 Searching and Sorting:

- Linear Search, Binary Search, Transpose sequential search, Binary search tree, Heap tree (application in priority queue and sorting), AVL tree, Splay tree, M-way search tree, B tree (insertion), B+ [25%]

tree (Definition and introduction), B* tree (Definition and introduction), Tries, Application of B tree and B+ tree in File Structures

- Hash Tables: Introduction, hash functions and hash keys, Collisions, Resolving collisions, Rehashing
- Sorting with algorithm analysis (best case, worst case, average): Bubble, Selection, Insertion, Shell, Merge, Quick, Heap, Radix

5 NP-Completeness and the P & NP Classes

- Introduction, Polynomial Time & Verification, NP-Completeness and Reducibility, The Vertex Cover Problem, The Traveling Salesman Problem, The Set Covering Problem [5%]

Notes:

Term work is to be carried out as per the above syllabus.

Data Structures to be implemented in any programming language

Main Reference Book(s):

- 1) Mark Allen Weiss, "Data Structures and Algorithm Analysis in C", Pearson Education, 2nd edition (2003)
- 2) G. A.V. PAI, "Data structures and algorithms, concepts, Techniques and Applications", TMH, 1st Edition (2008)

Suggested Additional Reading:

- 1) Horowitz, Sahni, Anderson-Freed, "Fundamentals of Data Structures in C", University Press (2nd edition-2007)
- 2) Jean-Paul Tremblay, Paul G. Sorenson, "An Introduction to Data Structures with Applications", Tata McGraw-Hill, 2nd Edition, (2007)
- 3) Cormen, Leiserson, Rivest, Stein, "Introduction to Algorithm", PHI (2003), 2nd Edition
- 4) Gilberg & Forouzan, "Data Structures: A Pseudo-code Approach with C", Thomson Learning.
- 5) Parag Dave & Himanshu Dave, "Design and Analysis of Algorithms", Pearson Education (2008)
- 6) Tanenbaum, "Data Structures Using C & C++", PHI.
- 7) Michel Goodrich, Roberto Tamassia, "Algorithm design-foundation, analysis & internet examples", Wiley
- 8) A V Aho, J E Hopcroft, J D Ullman, "Data Structures & Algorithms", Addison-Wesley Publishing (1983).
- 9) Michael Berman, "Data Structures Via C++: Objects by Evolution", Oxford Univ. Press (2004)
- 10) D E Knuth, "Sorting & Searching - The Art of Computer Programming", Vol. 3, Addison- Wesley Publishing (1973).

Chapter wise Coverage from Main Reference Book(s):

1	Book #1 -> Book #2 ->	Ch. 2, Section 2.1 Ch. 1, 2
2	Book #1 -> Book #2 ->	Ch. 3 (Section 3.2.1 to 3.2.7), 3.3 and 3.4 Ch. 3 (Section 3.4, 3.5), Ch.5 (Section 5.4, 5.5), Ch. 6 (Section 6.6), Ch. 7 (Section 7.5)
3	Book #1 -> Book #2 ->	Ch. 4 (Section 4.1, 4.2), Ch. 9 (Section 9.1 upto 9.6) Ch. 8 (Section 8.1 to 8.6, Section 8.7), Ch. 9 (Section 9.1 to 9.4)
4	Book #1 -> Book #2 ->	Ch. 4 (Section 4.3 to 4.5), Ch. 5 (Section 5.1 to 5.4) Ch. 6 (Section 6.1 to 6.3), Ch. 7 (Section 7.1 to 7.7) Ch. 10 (Section 10.4), Ch. 11 (Section 11.1 to 11.5), Ch. 13 (Section 13.6), Ch. 14 (Section 14.7, 14.8), Ch. 15 (Section 15.2, 15.3, 15.5) Ch. 16 (Section 16.2 upto 16.9)
5	Book #1 -> Reference Book #3 ->	Ch. 9 (Section 9.7) Ch. 34 (Section 34.1 to 34.3), Ch. 35 (Section 35.1 to 35.3)

Accomplishments of the student after completing the Course:

- Ability to decide the appropriate data type and data structure for a given problem.
 - Ability to select the best algorithm to solve a problem by considering various problem characteristics, such as the data size, the type of operations, etc.
 - The algorithms as referred above would include various operations on Queues, Stacks, Linked Lists, Trees, Graphs, Sorting, Searching, Hash tables
 - Ability to compare algorithms with respect to time and space complexity
-

Course Name: Object Oriented Concepts and Programming

Course Code: MCA122

Objectives:

The programming for small devices like mobile phones, networking devices like routers, coding for graphics and multimedia, requires efficient coding as well as object oriented programming. The C++ language fits perfectly as a tool for this type of work. How this important language is to be mastered and how to use this knowledge in building efficient and flexible code is one of the prime requirements today. The course presented here is targeting to enable the student to master such skills. Aim of the course is to enable students to

1. Differentiate between procedural and object oriented programming.
2. Learn C++ as a language and various features of it.
3. Learn Object Oriented principles and their application using C++.

Prerequisites:

1. Knowledge of C language
2. Programming concepts including algorithm building and logic

Contents :

1. Introduction to C++, Overview of Core C++ Language, Classes and Objects [10%]

Identifiers and constants (Literals), Keywords, Data Types, The Operators, New Casting Operators, typeid and throw, The Conditional structures and Looping Constructs, , The Difference between struct and class in C++,The difference between Union and Class, Static Data members of a class, Pointer to objects and pointer to members of class, The local classes, Assigning Objects

2. Functions [10%]

Introduction, The inline function, Default Arguments to the function,Functions with object as parameters,Call by reference and return by reference, Prototyping and Overloading, Friend functions, Const and Volatile functions, Static functions, Private and Public functions, Function Pointers, Adding C functions to the C++ program

3. Constructors and Destructors [10%]

Introduction to constructors, The explicit constructors, Parameterized constructors, Having multiple constructors, Constructors with default arguments, Dynamic Initialization, Constructor with dynamic allocation, copy constructors, The member initialization list, destructors

4. Operator Overloading and User Defined Conversions [10%]

Introduction, Unary Operators, Binary Operators, Using Friends as operator functions, Overloading other Operators, The need for user defined conversion, Four different cases where user defined conversions are needed, Comparison of both the methods of conversion

5. Templates [5%]

Function Templates, Non Generic (Non Type) Parameters in Template functions, Template function and specialization, Overloading a template function, Using Default Arguments, Class Templates, Classes with multiple generic data types, Static data members, Primary and Partial Specialization, The Export Keyword, The other use of typename

6. Inheritance [15%]

The need, Defining derived class using single base class, Derivation using public, private and protected access modifiers, The implementation of inheritance in the C++ object model, The Access Control, The Access Declaration, The multiple-inheritance, Abstract classes, Composite objects (container objects)

7. Runtime polymorphism by virtual functions [10%]

Compile Time and Runtime Polymorphism, Pointers to Objects, This pointer, Compatibility of Derived and base class pointers, The subobject concept, Virtual functions, Static invocation of virtual function, Default arguments to virtual functions, Virtual destructors, Pure virtual functions

8. IO Streams [5%]

Need for streams, Advantages of using C++ I/O over C IO, The C++ Predefined streams, Formatting IO, Formatting using ios members, Manipulators, Creating our own manipulator

9. Using Files for IO [10%]

Why IO is special, Text and binary streams, Opening and closing files, Dealing with text files Dealing with binary files, Providing Random Access using seek, IO Modes, Handling Errors

10. Namespaces [5%]

Introduction and need, Use the using syntax, Defining namespaces, Extending the namespace, Unnamed namespaces, Nested Namespaces, Namespace aliases, The std namespace, The Koenig lookup, Overhead with namespaces

11. The Standard Template Library [5%]

The STL (Standard Template Library) Introduction, Generic Programming , Generic Software Components, Generic Algorithms, Iterators, Containers, Algorithms

Main Reference Book:

1. Programming with ANSI C++ by Bhushan Trivedi, Oxford University Press

Suggested Additional Reading:

1. C++ FAQs by Pearson Education
2. C++ Primer by Stanley Lippmann Pearson Education
3. The C++ Programming Language by Bjarne Stroustrup, Pearson Education
4. Effective C++ by Scott Mayer Addison Wesley
5. Complete Reference C++ by Herbert Schildt McGraw Hill Publications

Chapter-wise coverage from main reference book :

Chapters : 1,2,3,4,5,6,7,9,10,12,13,14,16.

Accomplishments of the student after completing the course :

1. He/She should be able to understand and appreciate the Object Oriented approach of programming
2. He/She should be aware of the working and architectural model of C++.
3. He/She should be able to solve problems given to him/her using C++ with keeping balance between efficiency and flexibility

***** ***** *****

Course Name: Computer Oriented Numerical Methods

Course Code: MCA123

Objectives:

With the current deployment of computer technology and tools, it is very important to develop efficient algorithms for solving problems in science, engineering, technology, insurance & banking. Thus, the objective of this course is to enable students to obtain an intuitive and working understanding of numerical methods for the basic problems of numerical analysis and gain experience in the implementation of numerical methods using a computer. They would also gain an appreciation of the concept of error in these methods and the need to analyze and predict it.

Prerequisites: Basic knowledge of functions, logarithmic, trigonometric and exponential functions, graph of a function, polynomials, and roots of a polynomial, differentiation and integration.

Contents:

1. Computer Arithmetic [5%]

2. Iterative Methods [20%]

Bisection	}	Discuss convergence only without derivation
False Position		
Secant Method		
Newton Raphson		
Successive Approximation		
Birge Vieta Method		
Descarte's rule of sign		

3. Interpolation and Approximation [25%]

(a) **Polynomial interpolation:** Lagrange, forward difference, backward difference, divided difference interpolation, Error Estimates, Cubic spline interpolation, Inverse interpolation.

(b) **Approximation :** Least square Curve fitting: Linear Regression and Non linear Regression, Approximation of Functions by Taylor series, Chebyshev Approximation, Chebyshev Economization.

4. Numerical Differentiation & Integration [20%]

Differentiation
Integration - Trapezoidal, Simpson's 1/3 & 3/8 rules, Gauss Quadrature formulas

5. (a) Matrix [10%]

Introduction to Matrix, types of matrices, transpose of a matrix, matrix multiplication, Eigen values and Eigen vectors, Power method.

(b) Solution Of Simultaneous Linear Equations [10%]

Naïve Gauss Elimination, Gauss Elimination with pivoting, Gauss-Seidel method.

(c) Solution Of Ordinary Differential Equations[10%]

Taylor series, Runge-Kutta 2nd order, 3rd order, 4th order, Predictor Corrector methods : Milne Simpson & Adam's Moulton

Main Reference Books :

1. Numerical Methods for Engineers by Steven C Chapra & Raymond P Canale, Fifth Edition, Tata McGraw Hill Publication, Special Indian Edition.
2. Computer Oriented Numerical Methods by Dr. N Datta, Vikas Publication.

Suggested Additonal Reading :

1. Numerical Methods with Programs in C by T Veerarajan & T Ramachandran, Second Edition, Tata McGraw Hill Publication.
2. Numerical Methods by V. Rajaraman, Third Edition, Prentice-Hall India Pvt. Ltd.
3. Numerical Methods with C++ Programming by RM Somasundaram & RM Chandrasekaran, Prentice-Hall India Pvt. Ltd.
4. Applied Numerical Analysis by C F Gerald & P O Wheatley, Seventh Edition, Pearson Education Asia, New Delhi
5. Numerical Methods by Dr. V. N. Vedamurthy & Dr. N.Ch. S.N. Iyengar, Vikas Publication.
6. Numerical Analysis by Richard L. Burden, J. Douglas Faires, Cengage Publishcation.
7. Numerical Methods Using MATLAB by John H. Mathews, Kurtis D. Fink, Fourth Edition, Pearson Education

Chapter wise Coverage from the main reference books :

1 : From Book # 1 → Chp. 2 , Chp. 3 section 3.3 & section 3.4

2 : From Book # 1 → Chp. 4 , Chp. 5 upto subsection 5.3.2, Chp. 7 only section 7.1 (Case study)

From Book # 2 → Chp. 5 pages 118 – 123 (also refer suggested reading Bk # 1)

3: From Book # 1 → Chp. 12 section 12.1 except subsection 12.1.3, section 12.2, Chp. 13 only section 13.1 except 13.1.4, section 13.2, subsection 13.6.3 & subsection 13.6.4, Chp. 14 only section 1

From Book # 2 → Chp. 2 pages 17 – 23, 28 – 31 & 3

4 : From Book # 1 → Chp. 16 upto subsection 16.2.4, Chp. 17 section 17.3 except subsection 17.3.3, Chp. 18 only section 18.3

From Book # 2 → Chp. 3 pages 53 – 62

5: From Book # 1 → Chp. 8 upto section 8.4, Chp. 10 only section 10.2
Chp. 20 section 20.1 & section 20.3 upto 20.3.3
Chp. 21 only section 21.2.4

From Book # 2 → Chp. 6 pages 127 – 130 & 144 - 149

Accomplishments of the student after completing the course :

- ✓ Solve linear and non-linear algebraic equations, perform operations of calculus, fit curves, and solve differential equations using a computer
- ✓ Appreciate problems due to rounding errors and convergence

Course Name: Database Management Systems –II

Course Code : MCA124

Objectives: This course is intended to give students knowledge of how RDBMS is managed. It will prepare a theoretical as well as practical background of RDBMS.

Prerequisites : Database designing and retrieving using SQL.

Contents:

1. Query Processing and Optimization [25%]

Overview, Query interpretation, Equivalence of expressions, Join strategies for parallel processing, Algorithm for executing query operations, Heuristics of Query Optimization cost estimation of queries, Basic query optimization strategies: Selection operation, Sorting, Join operation

2. Transaction Processing and Concurrency control [30%]

Transaction concepts: Transaction execution and Problems, Transaction execution and control with SQL, Transaction properties, Transaction log, Concurrency control , Locking methods for concurrency control, Timestamp methods for concurrency control, Optimistic methods for concurrency control (Read phase, validation phase, Write phase), Deadlock handling - detection and resolution

3. Database backup and Recovery [25%]

Need of Database backup, Database backup techniques, Types of Database failures, Types of Database recovery (Forward recovery, Backward recovery, Media recovery), Recovery techniques (Deferred Update, Immediate update, Shadow Paging, Checkpoints), Buffer management.

4. Implementing Security in Databases [10%]

Security & integrity threats, Defense mechanisms, Statistical database auditing & control, Granting/revoking of privileges using SQL

5. Introduction to Other Databases [10%]

Overview of parallel databases, Overview of Distributed databases, Overview of Object oriented databases.

6. Procedural SQL Practical only

Understanding the main features of PL/SQL,PLSQL Architecture, advantages of using procedures, Basic code structure, variables, conditional statements, looping (loop statements, while loops, for loops, Cursor FOR loops) PL/SQL Stored procedures (functions, procedures, packages and triggers)

We have assumed that indexing and hashing will be covered in Data and File structures.

Main Reference Book(s):

1. Database Management Systems, Ramakrishnan, Gehrke, Third edition, McGraw Hill
2. SQL, PL/SQL – The programming Language Oracle-by Ivan Bayross

Suggested Additional Reading:

1. Database System Concept- Silberschatz, Korth, Sudarshan , Fifth Edition, McGraw Hill
2. Fundamentals of Database Systems, Elmasri Navathe, Third edition, Addison Wesley
3. Database Systems: Design , Implementation and Management , seventh edition , Peter Rob, Carlos Coronel, Thomson Course technology

Chapter wise Coverage from the main reference book(s):

Book No. 1 : Chapters -12 ,16,17,18,21,

Book No. 2 : Complete

Accomplishments of the student after completing the course:

Knowledge of handling multiple transactions effectively, Designing Stored procedures, utilization of triggers/cursors to control and retrieve data efficiently.

***** ***** *****

Course Name: Management Information System

Course Code: MCA125

Objectives:

We are living in the Information Society in the 21st century where Information is a critical component required at all levels of Management in an organization for its efficient functioning. Hence, every organization requires a System that would help in providing the required information support. Such a system is termed Management Information System. The students of MCA, once they successfully complete their studies, would be required to play a vital role in the development and implementation of MIS in their respective organizations. The objective of this course is to provide conceptual clarity to the students about functions of management, importance of information and, information needs at different levels, systems approach to the management of information, the process of development and implementation of MIS and finally examples of MIS required in different types of organizations. .

Prerequisites: ---

Contents:

Management Information System, Information Technology, Information Concepts, Systems Concepts, MIS support for Management, MIS Development and Implementation, MIS in functional areas in Manufacturing Organizations, MIS in Non-Manufacturing Organizations and New concepts like CRM, SCM, E business, M business etc. related to MIS, Introduction to IPR - Cyber Laws - Information Security, Case Study.

Main reference book(s):

Management Information Systems Text and Cases: by Dr. Milind Oka , Everest Publishing House, Pune, Thirteenth Edition.

Suggested Additional Reading:

1. Management Information Systems Text and Cases : by Waman S. Jawadekar, Tata McGraw Hill, Third Edition
2. Information Systems for Modern Management by Robert G. Murdick, Joel E. Ross, James R. Claggett , PHI, Third Edition

Chapter wise coverage from main reference book(s):

All 14 chapters of Main reference book (Management Information Systems by Dr. Milind Oka) **Except for following sections :**

Ch-2: 2.1 - 2.5, 2.11 - 2.26

Ch-4: 4.8 - 4.21

Ch-9: 9.1 – 9.9 (i.e whole chapter to be excluded)

Ch-14: 14.5

Introduction to IPR, Cyber Laws and Information Security to be covered from any latest book on MIS and should not exceed 2-3 pages per topic.

Accomplishment of the student after completing the course:

After completion of the course the student will be able to understand and appreciate the basic concepts of MIS, importance of MIS for an organization and will be able to contribute effectively in the development and implementation of MIS in different types of organizations.

Course Name : Theory of Computation

Course Code : MCA126

Objectives :

- Understanding and development of theoretical models of computations and their analysis.
- The models of computations include (i) Finite Automata (and Regular Languages), (ii) Push Down Automata (and Context-free Languages), (iii) Turing Machine (and their Languages).
- The aim of analysis is to identify and prove the capabilities and limitations of particular models of computations.
- Understanding that there are unsolvable problems that cannot be solved by any model of computation.
- Understanding of Complexity Theory, and Tractable and Intractable problems.

Prerequisites:

(a) Discrete Mathematics, (b) Mathematical Induction and Structural Induction

Contents :

1. Introduction, Sets , Logic , Functions , Relations , Languages , Proofs Mathematical Induction , Strong Principle of Mathematical Induction , Recursive Definitions , Structural Induction [7%]
2. Regular Languages & Regular Expressions, Finite Automata (FA), Distinguishing Strings w.r.t. Language , Union, Intersection, & Compliment of Languages [12%]
3. Non-deterministic Finite Automata (NFA) , NFA with Null-Transitions, Kleene's Theorem [12%]
4. A Criterion for Regularity, Minimal Finite Automata , Pumping Lemma for Regular Languages [12%]
5. Introduction to Context-Free Grammar (CFG) , Regular Grammars , Derivation (Parse) Trees & Ambiguities , An Unambiguous CFG for Algebraic Expressions , Simplified Forms & Chomsky Normal Forms [12%]
6. Introduction to Push Down Automata (PDA), Deterministic PDA (DPDA), PDA Corresponding to a Given CFG , CFG Corresponding to a Given PDA , Parsing [12%]
7. The Pumping Lemma for CFG , Intersection & Complement of CFGs , Decision Problems Involving CFGs [8%]
8. Turing Machine (TM) Definition & Examples, Computing a Partial Function with a TM, Combining TMs, Variations of TMs: Multi-tape TMs, Non-deterministic TMs, Universal TMs, Models of Computations & Church-Turing Thesis [12%]
9. Recursive Enumerable & Recursive Languages , Enumerating a Language , More General Grammars , Context-Sensitive Languages & Chomsky Hierarchy , Are All Languages Recursively Enumerable? [8%]
10. A Non-recursive Language & an Unsolvable Problem , The Halting Problem , Exercises. [5%]

Notes:

1. Only those proofs which use Induction, are included in the syllabus. In case of other theorems and Lemmas, proof may be omitted. However, the purpose, importance and applications of all theorems and Lemmas must be discussed.
2. Term work is to be carried out as per the above syllabus.

Main Reference Book:

John C. Martin, "Introduction to Languages and the Theory of Computation", Tata McGraw-Hill, (2003), 3rd Edition, ISBN: 007049939X

Suggested Additional Reading:

1. Harry Lewis & Christos H. Papadimitriou, "Elements of the Theory of Computation", 2nd Edition, EEE
2. Michael Sipser, "Introduction to the Theory of Computation", 2nd Edition, Boston, MA: Thomas Course Technology, 2006, ISBN: 0534950973
3. Parag Dave & Himanshu Dave, "Design and Analysis of Algorithms", Pearson Education
4. Cormen, Leiserson, Rivest, Stein, "Introduction to Algorithm", PHI (2003), 2nd Edition, Rs.325/-

Chapterwise Coverage from main reference book(s) :

Chapters : 1.1-1.5, 2.1-2.5, 3.1-3.5, 4.1-4.3, 5.1-5.3, 6.1-6.6, 7.1-7.6, 8.1-8.3, 9.1-9.7, 10.1-10.5, 11.1-11.2.

Accomplishment of Students after Studying this Course :

- Ability to distinguish between Regular Expressions and Non-regular Expressions.
- Ability to develop FA for a given regular language and vice versa (i.e. to develop a regular language from a given FA).
- Ability to use Pumping Lemma to optimize FA.
- Understanding of CFG with its potential to describe a big set of non-regular languages using finite number of production rules.
- Ability to develop Push Down Automata for a given CFG.
- Understanding of Turing Machine and their languages, with a good feel of the enhanced and bigger set of languages supported by Turing Machine.

***** ***** *****