# VIDYASAGAR UNIVERSITY

Midnapore, West Bengal

*PROPOSED CURRICULUM&SYLLABUS (DRAFT) OF*

## BACHELOR OF SCIENCE (HONOURS)

## MAJOR IN COMPUTER SCIENCE

### 4-YEAR UNDERGRADUATE PROGRAMME

*(w.e.f. Academic Year 2023-2024)*

***Based on***

**Curriculum & Credit Framework for Undergraduate Programmes**

**(CCFUP), 2023& NEP, 2020**

# VIDYASAGAR UNIVERSITY
# BACHELOR OF SCIENCE (HONOURS) MAJOR IN COMPUTER SCIENCE
# (under CCFUP, 2023)

| Level | YR. | SEM | Course Type | Course Code | Course Title | Credit | L-T-P | Marks | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | CA | ESE | TOTAL |
| B.Sc. (Hons.) | 3rd | V | SEMESTER-V | | | | | | | |
| | | | Major-8 | COSHMJ08 | T: Design and Analysis of Algorithms; P: Practical | 4 | 3-0-1 | 15 | 60 | 75 |
| | | | Major-9 | COSHMJ09 | T: Database Management Systems; P: Practical | 4 | 3-0-1 | 15 | 60 | 75 |
| | | | Major-10 | COSHMJ10 | P: Discrete Structure; | 4 | 3-1-0 | 15 | 60 | 75 |
| | | | Major Elective-01 | COSHDSE1 | Programming in Java / VB.NET / MATLAB Programming / ANDROID Programming | 4 | 3-1-0 / 3-0-1 | 15 | 60 | 75 |
| | | | Minor-5 (Disc.-I) | COSMIN05 | T: Operating System*; P:* Practical *(To be taken from other Discipline)* | 4 | 3-0-1 | 15 | 60 | 75 |
| | | | | | Semester-V Total | 20 | | | | 375 |
| | | VI | SEMESTER-VI | | | | | | | |
| | | | Major-11 | COSHMJ11 | T: Automata Theory; | 4 | 3-1-0 | 15 | 60 | 75 |
| | | | Major-12 | COSHMJ12 | T: Computer Graphics; P: Practical | 4 | 3-0-1 | 15 | 60 | 75 |
| | | | Major-13 | COSTHMJ13 | T: Microprocessor; P: Practical | 4 | 3-0-1 | 15 | 60 | 75 |
| | | | Major Elective-02 | COSHDSE2 | T: Internet of thing (IoT) / Block-Chain / Cloud Computing / Embedded System | 4 | 3-1-0/ 3-0-1 | 15 | 60 | 75 |
| | | | Minor-6 (Disc.-II) | COSMIN06 | T: Database Management Systems*; P:* Practical *(To be taken from other Discipline)* | 4 | 3-0-1 | 15 | 60 | 75 |
| | | | | | Semester-VI Total | 20 | | | | 375 |
| | | | | | YEAR-3 | 40 | | | | 750 |
| | | | Eligible to be awarded Bachelor of Science in Computer Science *on Exit* | | | 126 | Marks (Year: I+II+III) | | | 2325 |

MJ = Major, MI = Minor Course, DSE = Discipline Specific Elective Course, CA= Continuous Assessment, ESE= End Semester Examination, T = Theory, P= Practical, L-T-P = Lecture-Tutorial-Practical

## SEMESTER-V

### *MAJOR (MJ)*

**MJ-8: Design and Analysis of Algorithms** **Credits 04 (Full Marks: 75)**

**MJ8T: Design and Analysis of Algorithms** **Credits 03**

**OBJECTIVE OF THE COURSE**

- Equip students with the skills to create efficient algorithms and analyze their performance.
- Cover fundamental algorithmic techniques such as divide and conquer, dynamic programming, and greedy algorithms.
- Teach students to analyze time and space complexity of algorithms using Big O notation.
- Emphasize problem-solving strategies for designing optimal solutions to computational problems.
- Study data structures and their impact on algorithm efficiency.
- Provide practical experience through programming assignments and projects.
- Explore advanced topics like NP-completeness and approximation algorithms for addressing complex problems.
- Prepare students to evaluate and develop efficient algorithms, laying the foundation for careers in software development, data analysis, and research in computer science.

**Course Contents:**

**Module I : Basics of an Algorithm** **10 Hrs.**

Definition and Example of an algorithm, Characteristics of an algorithm, Steps in Designing of Algorithms, Growth of function, Recurrence, Problem Formulation (Tower of Hanoi),Substitution Method, Iteration Method, Master Method.

**Module II : Asymptotic Bounds** **10Hrs.**

Asymptotic Notations, Concept of efficiency of analysis of an algorithm Comparative efficiencies of algorithms: Linear, Quadratic, Polynomial and Exponential.

**Module III: Analysis of simple Algorithms** **10 Hrs.**

Euclid's algorithm for GCD, Horner's Rule for polynomial evaluation, Simple Matrix (n x n) Multiplication, Exponent evaluation e.g. an, Searching, Linear Search, Sorting, Bubble sort, Insertion Sort, Selection sort.

**Module IV: Greedy Technique** **5 Hrs.**

Elements of Greedy strategy, Activity Selection Problem, Continuous Knapsack Problem, Coin changing Problem.

**Module V: Divide and Conquer Approach** **15 Hrs.**

General Issues in Divide and Conquer, Binary Search, Merge Sort, Quick Sort, Integer Multiplication.

**Module VI: Graph Algorithm** **10 Hrs.**

Representation of Graphs, Adjacency Matrix, Adjacency List, Depth First Search and Examples, Breadth First Search and Examples

**References:**

1. T.H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein Introduction to Algorithms, PHI, 3rd Edition 2009
2. Sarabasse& A.V. Gelder Computer Algorithm – Introduction to Design and Analysis, Publisher – Pearson 3rd Edition 1999

**MJ8P: Design and Analysis of Algorithms Lab** **Credits 01**

1. Implement Insertion Sort (The program should report the number of comparisons).
2. Implement Heap Sort (The program should report the number of comparisons)
3. Implement Randomized Quick sort (The program should report the number of comparisons)
4. Implement Radix Sort
5. Create a Red-Black Tree and perform following operations on it:
   i. Insert a node , ii. Delete a node, iii. Search for a number & also report the color of the node containing this number.
6. Write a program to determine the LCS of two given sequences
7. Implement Breadth-First Search in a graph
8. Implement Depth-First Search in a graph
9. Write a program to determine the minimum spanning tree of a graph
10. Write a program to find minimum cost spanning tree using Kruskal's Algorithm.
11. Write a program to perform Single source shortest path problem for a given graph.
12. Write a program to find solution for job sequencing with deadlines problem.
13. Write a program for all pairs shortest path problem.
14. Write a program to solve N-QUEENS problem.
15. Write a program to solve Sum of subsets problem for a given set of distinct numbers.
16. Write a Program to perform Binary Search for a given set of integer values recursively and non-recursively.
17. Write a program to find solution for knapsack problem using greedy method.
18. Write a program to find minimum cost spanning tree using Prim's Algorithm.
19. Implement Merge Sort (The program should report the number of comparisons)

**MJ-9: Database Management Systems** **Credits 04 (FM: 75)**

**MJ-9T: Database Management Systems (Theory)** **Credits 03**

## OBJECTIVE OF THE COURSE

- Provide a comprehensive understanding of database design, implementation, and management.
- Cover fundamental concepts of database architecture, data models, and relational database management systems (RDBMS).
- Teach students to design and normalize databases to ensure data integrity and eliminate redundancy.
- Emphasize the use of SQL for querying and manipulating databases, including creating, updating, and deleting records.
- Explore advanced topics such as transactions, concurrency control, and database security for reliable data handling.
- Offer hands-on experience through projects involving the design and implementation of database systems.
- Equip students with skills to design, manage, and optimize databases.
- Prepare students for careers in database administration, data analysis, and software development.

## Course Contents:

**Module- I Introduction** **6 Hrs.**

Characteristics of database approach, data models, database system architecture and data independence

**Module- II Entity Relationship(ER) Modeling** **6 Hrs.**

Entity types, relationships, constraints.

**Module- III Relation data model** **10 Hrs.**

Relational model concepts, relational constraints normalization, relational algebra, SQL queries

**Module- IV Database design** **20 Hrs.**

Mapping ER/EER model to relational database, functional dependencies, Lossless decomposition, Normal forms (up to BCNF).

**Module- V Transaction Processing** **10 Hrs.**

ACID properties, concurrency control

**Module- VI File Structure and Indexing** **8 Hrs.**

Operations on files, File of Unordered and ordered records, overview of File organizations, Indexing structures for files( Primary index, secondary index, clustering index), Multilevel indexing using B and $B^+$ trees.

**Suggested Readings:**


1. R. Elmasri, S.B. Navathe, Fundamentals of Database Systems 6th Edition, Pearson Education, 2010.
2. R. Ramakrishanan, J. Gehrke, Database Management Systems 3rd Edition, McGraw-Hill, 2002.
3. A. Silberschatz, H.F. Korth, S. Sudarshan, DatabaseSystem Concepts 6th Edition, McGraw Hill, 2010.
4. R. Elmasri, S.B. Navathe Database Systems Models, Languages, Design and application Programming, 6th Edition, Pearson Education, 2013.


**MJ-9P: Database Management Systems (Practical) Credits 01**

**A. Create and use the following database schema to answer the given queries**

**EMPLOYEE Schema**

| Field | Type | NULL KEY | DEFAULT |
|---|---|---|---|
| **Eno** | Char(3) | PK | |
| Ename | Varchar(50) | | |
| Job_type | Varchar(50) | | Part |
| Manager | Char(3) | | |
| Hire_date | Date | | |
| Dno | Integer | FK | |
| Commission | Decimal(10,2) | | |
| Salary | Decimal(10,2) | | |

**DEPARTMENT Schema**

| Field | Type | NULL KEY | DEFAULT |
|---|---|---|---|
| Dno | Integer | PK | |
| Dname | Varchar(50) | NOT NULL | |
| Location | Varchar(50) | | KOLKATA |

**Query List**

1. Query to display Employee Name, Job, Hire Date, Employee Number; for each employee with the Employee Number appearing first.

2. Query to display unique Jobs from the Employee Table.
3. Query to display the Employee Name concatenated by a Job separated by a comma.
4. Query to display all the data from the Employee Table. Separate each Column by a comma and name the said column as THE_OUTPUT.
5. Query to display the Employee Name and Salary of all the employees earning more than $2850.

6. Query to display Employee Name and Department Number for the Employee No= 7900.
7. Query to display Employee Name and Salary for all employees whose salary is not in the range of $1500 and $2850.
8. Query to display Employee Name and Department No. of all the employees in Dept 10 and Dept 30 in the alphabetical order by name.
9. Query to display Name and Hire Date of every Employee who was hired in 1981.
10. Query to display Name and Job of all employees who don‘t have a current Manager.

**B.** Create a database having two tables with the specified fields, to computerize a library system of a Delhi University College.

**Library Books (Accession number, Title, Author, Department, PurchaseDate, Price)**
**Issued Books (Accession number, Borrower)**

a) Identify primary and foreign keys. Create the tables and insert at least 5 records ineach table.
b) Delete the record of book titled ―Database System Concepts‖.
c) Change the Department of the book titled ―Discrete Maths‖ to ―CS‖.
d) List all books that belong to ―CS‖ department.
e) List all books that belong to ―CS‖ department and are written by author ―Navathe‖.
f) List all computer (Department=‖CS‖) that have been issued.
g) List all books which have a price less than 500 or purchased between ―01/01/1999‖and ―01/01/2004‖.

**C.** Create a database having three tables to store the details of students of Computer Department in your college.

**Personal information about Student (College roll number, Name of student, Dateof birth, Address, Marks(rounded off to whole number) in percentage at 10 + 2, Phone number) Paper Details (Paper code, Name of the Paper)**

**Student’s Academic and Attendance details (College roll number, Paper code, Attendance, Marks in home examination).**

a. Identify primary and foreign keys. Create the tables and insert at least 5 records ineach table.
b. Design a query that will return the records (from the second table) along with the name of student from the first table, related to students who have more than 75% attendance and more than 60% marks in paper 2.
c. List all students who live in ―Delhi‖ and have marks greater than 60 in paper 1.
d. Find the total attendance and total marks obtained by each student.
e. List the name of student who has got the highest marks in paper 2.

**D.** Create the following tables and answer the queries given below:

**Customer (Cust ID, email, Name, Phone, Referrer ID) Bicycle (Bicycle ID, Date Purchased, Color, Cust ID,**
**Model No) Bicycle Model (Model No, Manufacturer,Style) Service (Start Date, Bicycle ID, End Date)**

a. Identify primary and foreign keys. Create the tables and insert at least 5 records ineach table.
b. List all the customers who have the bicycles manufactured by manufacturer
   i. ―Honda‖.
c. List the bicycles purchased by the customers who have been referred by customer
   i. ―C1‖.
d. List the manufacturer of red colored bicycles.
e. List the models of the bicycles given for service.

**E.** Create the following tables, enter at least 5 records in each table and answer the queriesgiven below.

**EMPLOYEE ( Person_Name, Street, City ) WORKS ( Person_Name, Company_Name, Salary ) COMPANY ( Company_Name, City ) MANAGES ( Person_Name, Manager_Name )**

a) Identify primary and foreign keys.
b) Alter table employee, add a column ―email‖ of type varchar (20).
c) Find the name of all managers who work for both Samba Bank and NCB Bank.
d) Find the names, street address and cities of residence and salary of all employees
   a. who work for ―Samba Bank‖ and earn more than $10,000.
e) Find the names of all employees who live in the same city as the company forwhich they work.
f) Find the highest salary, lowest salary and average salary paid by each company.
g) Find the sum of salary and number of employees in each company.
h) Find the name of the company that pays highest salary.

**F.** Create the following tables, enter at least 5 records in each table and answer the queriesgiven below.

**Suppliers (SNo, Sname, Status, SCity)Parts (PNo, Pname, Colour, Weight, City) Project (JNo, Jname, Jcity) Shipment (Sno, Pno, Jno, Qunatity)**

a) Identify primary and foreign keys.
b) Get supplier numbers for suppliers in Paris with status>20.
c) Get suppliers details for suppliers who supply part P2. Display the

supplier list inincreasing order of supplier numbers.

d) Get suppliers names for suppliers who do not supply part P2.
e) For each shipment get full shipment details, including total shipment weights.
f) Get all the shipments where the quantity is in the range 300 to 750 inclusive.
g) Get part nos. for parts that either weigh more than 16 pounds or are supplied bysuppliers S2, or both.
h) Get the names of cities that store more than five red parts.
i) Get full details of parts supplied by a supplier in London.
j) Get part numbers for part supplied by a supplier in London to a project in London.
k) Get the total number of project supplied by a supplier (say, S1).
l) Get the total quantity of a part (say, P1) supplied by a supplier (say, S1).

**MJ-10: Discrete Structure** **Credits 04(Full Marks: 75)**

**OBJECTIVE OF THE COURSE**

- Cover fundamental topics such as logic, set theory, combinatorics, graph theory, and discrete probability.
- Teach students to apply logical reasoning and proof techniques, including induction, to solve complex problems.
- Emphasize the use of mathematical structures to model and analyze real-world situations.
- Introduce algorithms for counting, permutations, and combinations to solve combinatorial problems.
- Develop skills in mathematical reasoning and problem-solving through practical exercises.
- Explore applications of discrete mathematics in cryptography, network design, and algorithms.
- Equip students with analytical tools for advanced studies in computer science, engineering, and mathematics.
- Prepare students for roles in research and development.

**MJ-10T: Discrete Structure (Theory)** **Credits 04**

**Course contents:**

**Module-I: Sets, Proof Templates, and Induction** **10Hrs.**

Basic Definitions, Operations on Sets, The Principle of Inclusion-Exclusion, Mathematical Induction, A Second Form of Induction.

**Module-II: Formal Logic** **6 Hrs.**

,

**Module-III: Relations** **6Hrs.**

Definitions and Operations, Special Types of Relations, Equivalence Relations, Ordering Relations.

**Module-IV: Functions** **6Hrs.**

Basic Definitions, Operations on Functions, The Pigeon-Hole Principle

**Module V: Recurrences:** **10 Hrs.**

Recurrence Relations, generating functions, Linear Recurrence Relations with constant coefficients and their solution, Substitution Method, Recurrence Trees, Master Theorem

**Module-VI: Analysis of Algorithms** **6Hrs.**

Comparing Growth Rates of Functions, Complexity of Programs

**Module-VII**: **Graph Theory** **10Hrs.**

Definitions, Connected Graphs ,The Kbnigsberg Bridge Problem ,Trees, Spanning Trees, Directed Graphs (Optional)

**Module-VIII**: **Counting and Combinatorics** **6Hrs.**

Counting Principles Permutations and Combinations Permutations and Combinations with Repetitions Combinatorial Identities (Optional) Pascal's Triangle (Optional)

**Recommended Books:**


1. C.L. Liu , D.P. Mahapatra, Elements of Discrete mathematics, 2nd Edition , Tata McGraw Hill, 1985,
2. Kenneth Rosen, Discrete Mathematics and Its Applications, Sixth Edition, McGraw Hill 2006
3. T.H. Coremen, C.E. Leiserson, R. L. Rivest, Introduction to algorithms, 3rd edition Prentice Hall on India, 2009
4. M. O. Albertson and J. P. Hutchinson, Discrete Mathematics with Algorithms, John wiley Publication, 1988
5. J. L. Hein, Discrete Structures, Logic, and Computability, 3rd Edition, Jones and Bartlett Publishers, 2009
6. D.J. Hunter, Essentials of Discrete Mathematics, Jones and Bartlett Publishers, 2008

## *MAJOR ELECTIVE (DSE)*

**Major Elective -1: Programming in Java** **Credits 04(Full Marks: 75)**

**MJ DSE-1T: Programming in Java (Theory)** **Credits 03**

**Course contents:**

**Introduction to Java** **8 Hrs.**

Java Architecture and Features, Understanding the semantic and syntax differences between C++ and Java, Compiling and Executing a Java Program, Variables, Constants, Keywords Data Types, Operators (Arithmetic, Logical and Bitwise) and Expressions, Comments, Doing Basic Program Output, Decision Making Constructs (conditional statements and loops) and Nesting, Java Methods (Defining, Scope, Passing and Returning Arguments, Type Conversion and Type and Checking, Built-in Java Class Methods),

**Arrays, Strings and I/O** **10 Hrs.**

Creating & Using Arrays (One Dimension and Multi-dimensional), Referencing Arrays Dynamically, Java Strings: The Java String class, Creating & Using String Objects, Manipulating Strings, String Immutability & Equality, Passing Strings To & From Methods, String Buffer Classes. Simple I/O using System out and the Scanner class, Byte and Character streams, Reading/Writing from console and files**.**

**Object-Oriented Programming Overview** **8 Hrs,**

Principles of Object-Oriented Programming, Defining & Using Classes, Controlling Access to Class Members, Class Constructors, Method Overloading, Class Variables & Methods, Objects as parameters, final classes, Object class, Garbage Collection.

**Inheritance, Interfaces, Packages, Enumerations, Autoboxing and Metadata** **10 Hrs.**

Inheritance: (Single Level and Multilevel, Method Overriding, Dynamic Method) Dispatch, Abstract Classes), Interfaces and Packages, Extending interfaces and packages, Package and Class Visibility, Using Standard Java Packages (util, lang, io, net), Wrapper Classes, Autoboxing/Unboxing, Enumerations and Metadata.

**Exception Handling, Threading, Networking and Database Connectivity** **10 Hrs.**

Exception types, uncaught exceptions, throw, built-in exceptions, Creating your own exceptions; Multi-threading: The Thread class and Runnable interface, creating single and multiple threads, Thread prioritization, synchronization and communication, suspending/resuming threads. Using java.net package, Overview of TCP/IP and Datagram programming. Accessing and manipulating databases using JDBC.

**Applets and Event Handling** **10 Hrs.**

Java Applets: Introduction to Applets, Writing Java Applets, Working with Graphics, Incorporating Images & Sounds. Event Handling Mechanisms, Listener Interfaces, Adapter and Inner Classes. The design and Implementation of GUIs using the AWT controls, Swing components of Java Foundation Classes such as labels, buttons, textfields, layout managers, menus, events and listeners; Graphic objects for drawing figures such as lines, rectangles, ovals, using different fonts. Overview of servlets.

**Reference Books**


1. Ken Arnold, James Gosling, David Homes, "The Java Programming Language", 4th Edition, 2005.
2. James Gosling, Bill Joy, Guy L Steele Jr, Gilad Bracha, Alex Buckley "The Java Language Specification, Java SE 8 Edition (Java Series)", Published by Addison Wesley, 2014.
3. Joshua Bloch, "Effective Java" 2nd Edition, Publisher: Addison-Wesley, 2008.
4. Cay S. Horstmann, Gary Cornell, "Core Java 2 Volume 1 ,9th Edition, Printice Hall.2012
5. Cay S. Horstmann, Gary Cornell, "Core Java 2 Volume 2 - Advanced Features)", 9th Edition, Printice Hall.2013
6. Bruce Eckel, "Thinking in Java", 3rd Edition, PHI, 2002.
7. E. Balaguruswamy, "Programming with Java", 4th Edition, McGraw Hill.2009.
8. Paul Deitel, Harvey Deitel, "Java: How to Program", 10th Edition, Prentice Hall, 2011.
9. "Head First Java", Orielly Media Inc. 2nd Edition, 2005.
10. David J. Eck, "Introduction to Programming Using Java", Published by CreateSpace Independent Publishing Platform, 2009.
11. John R. Hubbard, "Programming with JAVA", Schaum's Series, 2nd Edition, 2004.


## MJ DSE-1P: Programming in Java Lab (Practical) Credits 01

**Course Outline:**

1. To find the sum of any number of integers entered as command line arguments
2. To find the factorial of a given number
3. To learn use of single dimensional array by defining the array dynamically.
4. To learn use of lenth in case of a two dimensional array
5. To convert a decimal to binary number
6. To check if a number is prime or not, by taking the number as input from the keyboard
7. To find the sum of any number of integers interactively, i.e., entering every number from the keyboard, whereas the total number of integers is given as a command line argument
8. Write a program that show working of different functions of String and String Buffer classs like set Charat ( set Length (), append (), insert (), concat ()and equals ().
9. Write a program to create a class with methods where distance is computed in terms of feet and inches, how to create objects of a class and to see the use of this pointer
10. Modify the class by creating constructor for assigning values (feet and inches) to the distance object. Create another object and assign second object as reference variable to another object reference variable. Further create a third object which is a clone of the first object.
11. Write a program to show that during function overloading, if no matching argument is found, then java will apply automatic type conversions(from lower to higher data type)

12. Write a program to show the difference between public and private access specifiers. The program should also show that primitive data types are passed by value and objects are passed by reference and to learn use of final keyword
13. Write a program to show the use of static functions and to pass variable length arguments in a function.
14. Write a program to demonstrate the concept of boxing and unboxing.
15. Create a multi-file program where in one file a string message is taken as input from the user and the function to display the message on the screen is given in another file (make use of Scanner package in this program).
16. Write a program to create a multilevel package and also creates a reusable class to generate Fibonacci series, where the function to generate fibonacii series is given in a different file belonging to the same package.
17. Write a program that creates illustrates different levels of protection in classes/subclasses belonging to same package or different packages
18. Write a program that takes two numbers a and b as input, computes a/b, and invokes Arithmetic Exception to generate a message when the denominator is zero.
19. Write a program to show the use of nested try statements that emphasizes the sequence of checking for catch handler statements.
20. Write a program to create your own exception types to handle situation specific to your application (Hint: Define a subclass of Exception which itself is a subclass of Throwable).
21. Write a program to demonstrate priorities among multiple threads.
22. Write a program to demonstrate multithread communication by implementing synchronization among threads (Hint: you can implement a simple producer and consumer problem).
23. Write a program to create URL object, create a URL Connection using the open Connection () method and then use it examine the different components of the URL and content.
24. Write a program to implement a simple datagram client and server in which a message that is typed into the server window is sent to the client side where it is displayed.
25. Write a program that creates a Banner and then creates a thread to scrolls the message in the banner from left to right across the applet‘s window.
26. Write a program to get the URL/location of code (i.e. java code) and document (i.e. html file).
27. Write a program to demonstrate different mouse handling events like mouse Clicked (), mouse Entered (), mouse Exited (), mouse Pressed, mouse Released () and mouse Dragged ().
28. Write a program to demonstrate different keyboard handling events.
29. Write a program to generate a window without an applet window using main () function.
30. Write a program to demonstrate the use of push buttons.

**OR**

**Major Elective -1: VB.NET** **Credits 04 (Full Marks: 75)**

**MJ DSE-1T: VB.NET (Theory)** **Credits 03**

**Course contents:**

**GUI Environment:** **6 Hrs.**

Introduction to graphical user interface (GUI), programming language (procedural, object oriented, event driven), the GUI environment, compiling, debugging, and running the programs.

**Controls:** **8 Hrs.**

Introduction to controls textboxes, frames, check boxes, option buttons, images, setting orders and styles, the shape control, the line control, working with multiple controls and their properties, designing the user interface, keyboard access, tab controls, default & cancel property, coding for controls

**Operations:** **6 Hrs.**

Data types, constants, named & intrinsic, declaring variables, scope of variables, val function, arithmetic operations, formatting data.

**Decision Making:** **8 Hrs.**

If statement, comparing strings, compound conditions (and, or, not), nested if statements, case structure, using if statements with option buttons & check boxes, displaying message in message box, testing whether input is valid or not.

**Modular programming:** **8 Hrs.**

Menus, sub-procedures and sub-functions defining / creating and modifying a menu, using common dialog box, creating a new sub- procedure, passing variables to procedures, passing argument by value or by reference, writing a function/ procedure.

**Forms Handling:** **6 Hrs.**

Multiple forms creating, adding, removing forms in project, hide, show method, load, unloadstatement, me keyword, referring to objects on a different forms.

**Iteration Handling:** **4 Hrs.**

Do/loops, for/next loops, using msgbox function, using string function

**Arrays and Grouped Data Control:** **8 Hrs.**

Arrays - 1-dimension arrays, initializing an array using for each, user-defined data types, accessing information with user-defined data types, using list boxes with array, two dimensional arrays. lists, loops and printing list boxes & combo boxes, filling the list using property window / add item method, clear method, list box properties, removing an item from a list, list box/ combo box operations.

**Database Connectivity:** **6 Hrs.**

Database connectivity of forms with back end tool like mysql, populating the data in text boxes, listboxes etc. searching of data in database using forms. Updating/ editing of data based on a criterion

**Suggested Readings:**

1. Programming in Visual Basic 6.0 by Julia Case Bradley, Anita C. Millispangh (Tata Mcgraw Hill Edition 2000 (Fourteenth Reprint 2004))

**MJ DSE-1P: VB.NET (Practical)** **Credits 01**

1. Print a table of numbers from 5 to 15 and their squares and Cubes.
2. Print the largest of three numbers.
3. Find the factional of a number n.
4. Enter a list of positive numbers terminated by zero. Find the sum and average of these numbers.
5. A person deposits Rs. 1000 in a fixed account yielding 5% interest. Complete the amount in the account at the end of each year for n years.
6. Read n numbers. Count the number of negative numbers, positive numbers and zeros in the list.
7. Read n numbers. Count the number of negative numbers, positive numbers and zeroes in the list. use arrays.
8. Read a single dimension array. Find the sum and average of these numbers.
9. Read a two dimension array. Find the sum of two 2D Array.
10. Create a database Employee and Make a form in VB 6.0 to allow data entry to Employee Form.

**OR**

**Major Elective -1: MATLAB Programming** **Credits 04 (Full Marks: 75)**

**MJ DSE-1T: MATLAB Programming (Theory)** **Credits 03**

**Course contents:**

**Introduction to Programming**: Components of a computer, working with numbers, Machine code, Software hierarchy. **8 Hrs.**

**Programming Environment**: MATLAB Windows, A First Program, Expressions, Constants, Variables and assignment statement, Arrays. **8 Hrs.**

**Graph Plots**: Basic plotting, Built in functions, Generating waveforms, Sound replay, load and save. **8 Hrs.**

**Procedures and Functions**: Arguments and return values, M-files, Formatted console input-output, String handling. **10Hrs.**

**Control Statements**: Conditional statements: If, Else, Else-if, Repetition statements: While, for loop. **8 Hrs.**

**Manipulating Text**: Writing to a text file, Reading from a text file, Randomising and sorting a list, searching a list. **8 Hrs.**

**GUI Interface**: Attaching buttons to actions, Getting Input, Setting Output. **10Hrs.**

**Recommended Books:**


1. MATLAB: An Introduction with Applications, by Amos Gilat, 2nd edition, Wiley, 2004,
2. C.B. Moler, Numerical Computing with MATLAB, SIAM, 2004.


**MJ DSE-1P: MATLAB Programming (Practical)** **Credits 01**

1. Write a program to assign the following expressions to a variable A and then to print out the value of A.
   a) (3+4)/(5+6)
   b) $2\pi^2$
   c) $\sqrt{2}$
   d) $(0.0000123 + 5.67 \times 10^{-3}) \times 0.4567 \times 10^{-4}$

2. Celsius temperatures can be converted to Fahrenheit by multiplying by 9, dividing by 5, and adding 32. Assign a variable called C the value 37, and implement this formula to assign a variable F the Fahrenheit equivalent of 37 Celsius.
3. Set up a vector called N with five elements having the values: 1, 2, 3, 4, 5. Using N,

create assignment statements for a vector X which will result in X having these values:

a. 2, 4, 6, 8, 10
b. 1/2, 1, 3/2, 2, 5/2
c. 1, 1/2, 1/3, 1/4, 1/5
d. 1, 1/4, 1/9, 1/16, 1/25

4. A supermarket conveyor belt holds an array of groceries. The price of each product (in pounds) is [ 0.6, 1.2 ,0.5, 1.3 ] ; while the numbers of each product are [ 3, 2 ,1 ,5 ]. Use MATLAB to calculate the total bill.
5. The sortrows(x) function will sort a vector or matrix X into increasing row order. Use this function to sort a list of names into alphabetical order.
6. The ―identity‖ matrix is a square matrix that has ones on the diagonal and zeros elsewhere. You can generate one with the eye() function in MATLAB. Use MATLAB to find a matrix B, such that when multiplied by matrix A=[ 1 2; -1 0 ] the identity matrix I=[ 1 0; 0 1 ] is generated. That is A*B=I.
7. Create an array of N numbers. Now find a single MATLAB statement that picks out from that array the 1,4,9,16,…,√Nth entries, i.e. those numbers which have indices that are square numbers.
8. Draw a graph that joins the points (0,1), (4,3), (2,0) and (5,-2).
9. Calculate 10 approximate points from the function y=2x by using the formulae:
   xn = n
   yn = 2n + rand - 0.5
10. Fit a line of best fit to these points using the function polyfit() with degree=1, and generate co-ordinates from the line of best fit using polyval(). Use the on-line help to find out how to use these functions. Plot the raw data and the line of best fit.
11. Calculate and replay 1 second of a sinewave at 500Hz with a sampling rate of 11025Hz. Save the sound to a file called "ex35.wav". Plot the first 100 samples.
12. Calculate and replay a 2 second chirp. That is, a sinusoid that steadily increases in frequency with time, from say 250Hz at the start to 1000Hz at the end.
13. Build a square wave by adding together 10 odd harmonics: 1f, 3f, 5f, etc. The amplitude of the nth harmonic should be 1/n. Display a graph of one cycle of the result superimposed on the individual harmonics.
14. Write a function called FtoC (ftoc.m) to convert Fahrenheit temperatures into Celsius. Make sure the program has a title comment and a help page.
15. Write a program to input 2 strings from the user and to print out (i) the concatenation of the two strings with a space between them, (ii) a line of asterisks the same length as the concatenated strings, and (iii) the reversed concatenation. For example:
    Enter string 1: Mark
    Enter string 2: Huckvale
    Mark Huckvale
    **********

## OR

**Major Elective -1: ANDROID Programming** **Credits 04 (Full Marks: 75)**

**MJ DSE-1T: ANDROID Programming (Theory)** **Credits 03**

**Course contents:**

**Introduction:** History of Android, Introduction to Android Operating Systems, Android Development Tools, Android Architecture. **5 Hrs.**

**Overview of object oriented programming using Java**: OOPs Concepts: Inheritance, Polymorphism, Interfaces, Abstract class, Threads, Overloading and Overriding, Java Virtual Machine. **15 Hrs.**

**Development Tools:** Installing and using Eclipse with ADT plug-in, Installing Virtual machine for Android sandwich/Jelly bean (Emulator), configuring the installed tools, creating a android project – Hello Word, run on emulator, Deploy it on USB-connected Android device. **15 Hrs**.

**User Interface Architecture**: Application context, intents, Activity life cycle, multiple screen sizes. **9 Hrs**.

**User Interface Design:** Form widgets, Text Fields, Layouts, Button control, toggle buttons, Spinners(Combo boxes),Images, Menu, Dialog. **8 Hrs.**

**Database**: Understanding of SQLite database, connecting with the database. **8 Hrs.**

**Book Recommended:**

1. Android application development for java programmers. By James C. Sheusi. Publisher: Cengage Learning, 2013.

***Online Reading / Supporting Material:***

1. http://www.developer.android.com
2. http://developer.android.com/about/versions/index.html
3. http://developer.android.com/training/basics/firstapp/index.html
4. http://docs.oracle.com/javase/tutorial/index.htm (Available in the form of free downloadable ebooks also).
5. http://developer.android.com/guide/components/activities.html
6. http://developer.android.com/guide/components/fundamentals.html
7. http://developer.android.com/guide/components/intents-filters.html.
8. http://developer.android.com/training/multiscreen/screensizes.html
9. http://developer.android.com/guide/topics/ui/controls.html

10. http://developer.android.com/guide/topics/ui/declaring-layout.html
11. http://developer.android.com/training/basics/data-storage/databases.html

**MJ DSE-1P: ANDROID Programming Lab (Practical) Credits 01**

1. Create ―Hello World‖ application. That will display ―Hello World‖ in the middle ofthe screen in the emulator. Also display ―Hello World‖ in the middle of the screen inthe Android Phone.
2. Create an application with login module. (Check username and password).
3. Create spinner with strings taken from resource folder (res >> value folder) and on changing the spinner value, Image will change.
4. Create a menu with 5 options and and selected option should appear in text box.
5. Create a list of all courses in your college and on selecting a particular course teacher-in-charge of that course should appear at the bottom of the screen.
6. Create an application with three option buttons, on selecting a button colour of thescreen will change.
7. Create and Login application as above. On successful login, pop up the message.
8. Create an application to Create, Insert, update, Delete and retrieve operation on the database.

## *MINOR (MI)*

**Minor (MI) -5: Operating System** **Credits 04 (Full Marks: 75)**

**OBJECTIVE OF THE COURSE**

- Provide a comprehensive understanding of fundamental concepts and functions of modern operating systems.
- Cover the architecture and components of operating systems, including process management, memory management, file systems, and input/output systems.
- Teach about concurrency, process synchronization, and inter-process communication for efficient task management.
- Emphasize the role of operating systems in resource allocation and system security.
- Offer hands-on experience in implementing and configuring operating system features through labs and projects.
- Explore various operating systems like Windows, Linux, and macOS to understand their differences and commonalities.
- Delve into virtualization and distributed systems to highlight current trends in operating system design.
- Equip students with the skills to analyze, design, and optimize operating systems, preparing them for advanced study and careers in systems programming and software engineering.

**MI -5T: Operating System (Theory)** **Credits 03**

**Course contents:**

**Module I: Introduction** **10 Hrs.**

Basic OS functions, resource abstraction, types of operating systems–multiprogrammingsystems, batch systems , time sharing systems; operating systems for personal computers & workstations, process control & real time systems.

Case study on Linux system **6 Hrs.**

- Cloud computing (3 lectures)
- Linux evolution and Linux distros (2 lectures)
- Linux file system (1 lecture)

**Module II: Operating System Organization** **6 Hrs.**

Processor and user modes, kernels, system calls and system programs.

**Module III: Process Management** **16 Hrs.**

System view of the process and resources, process abstraction, process hierarchy, threads, threading issues, thread libraries; Process Scheduling, non-pre-emptive and pre-emptive scheduling algorithms; concurrent processes, critical section, semaphores, methods for inter-process communication; deadlocks.

**Module IV: Memory Management** **12 Hrs.**

Physical and virtual address space; memory allocation strategies – fixed and variablepartitions, paging, segmentation, virtual memory

**Module V: I/O Management** **10 Hrs.**

Directory structure, file operations, file allocation methods, device management.

**Recommended Books:**


1. A Silberschatz, P.B. Galvin, G. Gagne, Operating Systems Concepts, 8th Edition,John Wiley Publications 2008.
2. A.S. Tanenbaum, Modern Operating Systems, 3rd Edition, Pearson Education 2007.
3. G. Nutt, Operating Systems: A Modern Perspective, 2nd Edition Pearson Education1997.
4. W. Stallings, Operating Systems, Internals & Design Principles , 5th Edition, Prentice Hall of India. 2008.
5. M. Milenkovic, Operating Systems- Concepts and design, Tata McGraw Hill 1992.


**MI -5P: Operating System Lab (Practical)** **Credits 01**

1. Write a program (using *fork ()* and/or *exec ()* commands) where parent and childexecute:
   a) same program, same code.
   b) same program, different code.
   c) before terminating, the parent waits for the child to finish its task.
2. Write a program to report behaviour of Linux kernel including kernel version, CPUtype and model. (CPU information)
3. Write a program to report behaviour of Linux kernel including information onconfigured memory, amount of free and used memory (memory information).

4. Write a program to print file details including owner access permissions, file accesstime, where file name is given as argument.
5. Write a program to copy files using system calls.
6. Write program to implement FCFS scheduling algorithm.
7. Write program to implement Round Robin scheduling algorithm.
8. Write program to implement SJF scheduling algorithm.
9. Write program to calculate sum of n numbers using *thread* library.
10. Write a program to implement first-fit, best-fit and worst-fit allocation strategies

# SEMESTER-VI

## *MAJOR (MJ)*

**MJ-11: Automata Theory** **Credits 04 (Full Marks: 75)**

**OBJECTIVE OF THE COURSE**

- Cover essential concepts such as finite automata, regular expressions, context-free grammars, and Turing machines.
- Teach students to analyze the capabilities and limitations of different computational models.
- Emphasize the connection between automata and formal languages, including designing automata for language recognition.
- Explore principles of computability and complexity, including decidability and the classes of P and NP problems.
- Develop problem-solving skills through exercises and projects that apply theoretical concepts to practical scenarios.
- Highlight the role of automata theory in compiler design, algorithm development, and artificial intelligence.
- Equip students with analytical tools to tackle complex computational problems.
- Prepare students for advanced studies and research in computer science and related fields.

**MJ-11T: Automata Theory (Theory)** **Credits 03**

**Course contents:**

**Module I: Languages** **10 Hrs.**

Alphabets, string, language, Basic Operations on language, Concatenation, KleeneStar

**Module II: Finite Automata and Regular Languages** **15 Hrs**

Regular Expressions, Transition Graphs, Deterministic and non-deterministic finite automata, NFA to DFA Conversion, Regular languages and their relationship with finite automata, Pumping lemma and closure properties of regular languages.

**Module III: Context free languages** **20 Hrs.**

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

**Module IV. Turing Machines and Models of Computations 15 Hrs.**

RAM, Turing Machine as a model of computation, Universal Turing Machine, Language acceptability, decidability, halting problem, Recursively enumerable and recursive language unsolvability problems.

**Suggested Readings:**


1. Daniel I.A.Cohen, Introduction to computer theory, John Wiley,1996
2. Lewis & Papadimitriou, Elements of the theory of computation, PHI 1997.
3. Hoperoft, Aho, Ullman, Introduction to Automata theory, Language & Computation, 3rd Edition, Pearson Education. 2006
4. P. Linz, An Introduction to Formal Language and Automata 4th edition Publication Jone Bartlett, 2006

**MJ-12: Computer Graphics** **Credits 04 (Full Marks: 75)**

**OBJECTIVE OF THE COURSE**

- Understand the principles and techniques used in generating and manipulating visual content.
- Cover fundamental concepts such as rasterization, vector graphics, and 3D modeling.
- Learn about graphic transformations, rendering techniques, and the mathematics behind image generation.
- Emphasize the use of algorithms for image processing, animation, and visualization.
- Gain hands-on experience with graphic design tools and programming libraries like OpenGL and DirectX.
- Explore advanced topics such as ray tracing, shading models, and GPU programming.
- Develop skills to create and manipulate digital images and animations.
- Prepare for careers in game development, animation, and multimedia design.

**MJ-12T: Computer Graphics (Theory)** **Credits 03**

**Course contents:**

**Module I: Basics of Computer Graphics** **6 Hrs.**

Introduction, What is computer Graphics?, Area of Computer Graphics, Design and Drawing, Animation Multimedia applications, Simulation, How are pictures actually stored and displayed, Difficulties for displaying pictures.

**Module II: Graphic Devices** **6 Hrs.**

Cathode Ray Tube, Quality of Phosphors, CRTs for Color Display, Beam Penetration CRT, The Shadow - Mask CRT, Direct View Storage Tube, Tablets, The light Pen, Three Dimensional Devices.

**Module III: C Graphics Basics** **2 Hrs.**

Graphics programming, initializing the graphics, C Graphical functions, simple Programs

**Module IV: Simple Line Drawing Methods** **10 Hrs.**

Point Plotting Techniques, Qualities of good line drawing algorithms, The Digital Differential Analyzer (DDA), Bresenham's Algorithm, Generation of Circles

**Module V: Two Dimensional Transformations and Clipping and Windowing** **10 Hrs.**

What is transformation?, Matrix representation of points, Basic transformation, Need for Clipping and Windowing, Line Clipping Algorithms, The midpoint subdivision Method, Other Clipping Methods, Sutherland - Hodgeman Algorithm, Viewing Transformations

**Module VI: Three Dimensional Graphics 10 Hrs.**

Need for 3-Dimensional Imaging, Techniques for 3-Dimesional displaying, Parallel Projections, Perspective projection, Intensity cues, Stereoscope effect, Kinetic depth effect, Shading

**Module VII: Solid Area Scan Conversion and Three Dimensional Transformations 8 Hrs.**

Solid Area Scan Conversion, Scan Conversion of Polygons, Algorithm Singularity, Three Dimensional transformation, Translations, Scaling, Rotation, Viewing Transformation, The Perspective, Algorithms, Three Dimensional Clipping, Perspective view of Cube.

**Module VIII: Hidden Surface Removal 8 Hrs.**

Need for hidden surface removal, The Depth - Buffer Algorithm, Properties that help in reducing efforts, Scan Line coherence algorithm, Span - Coherence algorithm, Area-Coherence Algorithms, Warnock's Algorithm, Priority Algorithms

**Suggested Readings:**


1. J.D.Foley, A.Van Dan, Feiner, Hughes Computer Graphics Principles & Practice 2nd edition Publication Addison Wesley 1990.
2. D.Hearn, Baker: Computer Graphics, Prentice Hall of India 2008.
3. D.F.Rogers Procedural Elements for Computer Graphics, McGraw Hill 1997.
4. D.F.Rogers, Adams Mathematical Elements for Computer Graphics, McGraw Hill 2nd edition1989.


**MJ-12P: Computer Graphics Lab (Practical) Credits 01**

1. Write a program to implement Bresenham's line drawing algorithm.
2. Write a program to implement mid-point circle drawing algorithm.
3. Write a program to clip a line using Cohen and Sutherland line clipping algorithm.
4. Write a program to clip a polygon using Sutherland Hodgeman algorithm.
5. Write a program to apply various 2D transformations on a 2D object (use homogenous coordinates).
6. Write a program to apply various 3D transformations on a 3D object and then apply parallel and perspective projection on it.
7. Write a program to draw Hermite/Bezier curve.
8. Create and rotate a triangle about the origin and a fixed point.
9. Draw a color cube and allow the user to move the camera suitably to experiment with

perspective viewing.

10. To draw a simple shaded scene consisting of a tea pot on a table.
11. Develop a menu driven program to fill the polygon using scan line algorithm
12. Implementation of polygon filling using Flood-fill, Boundaryfill and Scan-line algorithms.
13. Implementation of 2D transformation: Translation, Scaling,
14. Rotation, Mirror Reflection and Shearing (write a menu driven program).
15. Implementation of Line Clipping using Cohen-Sutherland algorithm and Bisection Method.
16. Implementation of 3D geometric transformations: Translation, Scaling and rotation.
17. Implementation of Curve generation using Interpolation methods.
18. Implementation of Curve generation using B-spline and Bezier curves.
19. Write a program to draw a Hut or other geometrical figures.
20. Write a program to draw a line using DDA algorithm.
21. Write a program to draw a line using Mid-Point algorithm.
22. Write a program to draw an Ellipse using Mid-Point algorithm.
23. Write a program to rotate a Circle around any arbitrary point or around the boundary of another circle.
24. Write a menu driven program to rotate, scale and translate a line point, square, triangle about the origin.
25. .Write a program to perform line clipping.
26. .Write a program to implement reflection of a point, line.
27. .Write a program to perform shearing on a line.
28. .Write a program to implement polygon filling.

**MJ-13: Microprocessor** **Credits 04 (Full Marks: 75)**

**OBJECTIVE OF THE COURSE**

- Provide a comprehensive understanding of microprocessor architecture and its applications in computing.
- Cover fundamental concepts of microprocessor operations, including instruction sets, data paths, and control signals.
- Teach assembly language programming to develop low-level software for microprocessors.
- Emphasize interfacing techniques to connect microprocessors with memory and peripheral devices.
- Explore various microprocessor families and their characteristics, such as Intel and ARM architectures.
- Offer hands-on experience in programming and debugging microprocessor-based systems through practical lab sessions.
- Delve into advanced topics like microcontroller applications, embedded systems, and real-time processing.
- Equip students with skills to design and implement microprocessor-based solutions, preparing them for careers in embedded systems design, hardware engineering, and electronics development.

**MJ-13T: Microprocessor (Theory)** **Credits 03**

**Course contents:**

**Module-I: Introduction to microprocessors** **6 Hrs.**

Features, Programmers model, external & internal organization.

**Module -II: 8085 Architecture** **10 Hrs.**

8085 Architecture & organization, Instruction cycles, machine cycles and T-states, address decoding Techniques, minimum system design, Memory interfacing with timing considerations, clock, reset & buffering circuits . 8085 Instruction set: Instruction format, addressing modes, classification of instruction set, 8085 assembly language programming: basic structure, data transfer, arithmetical, logical, transfer of control & Miscellaneous instruction types.

**Module -III: Stack & subroutines** **8 Hrs.**

Stack operations, limitations, subroutine concepts, parameter passing techniques, subroutine design, delay subroutine design & applications, Re-entrant & recursive subroutines, concept of counters and timers.

**Module -IV: I/O data transfer techniques** **8 Hrs.**

I/O interface concepts, speed considerations, program controlled I/O, asynchronous & synchronous I/O techniques interrupt driven program controlled I/O, direct memory access data control techniques, handshake signals, concepts of serial communication, matrix keyboard &multiplexed display interface.

**Module -V: Interrupts** **10 Hrs.**

Requirements, single level interrupt; multilevel interrupt & vector interrupt system, 8085interrupt structure and its operation, 8259A interrupt controller.

**Module -VI: I/O controllers** **8 Hrs.**

Features, organization & operating modes of 8155 multifunction device,8255programmable peripheral interface,8254 programmable timer,8237 programmable DMA controller.

**Module -VII: 16 bit processors** **8Hrs.**

8086 and architecture, segmented memory has cycles, read/write cycle in min/max mode. Reset operation, wait state, Halt state, Hold state, Lock operation, interrupt processing.

**MJ-13P: Microprocessor Lab (Practical)** **Credits 01**

1. Write an assembly language program to add two 8-bit numbers in 8085.
2. Write a program using 8085 to store 16 bit of data in memory location at 2050 to 205F. Transfer the entire block of data to new memory location starting at 2070H.
3. Write an assembly language program to subtract two 8-bit numbers.
4. Write an assembly language program to add two 16-bits numbers.
5. Write an assembly language program to add two 16-bits number with ADD and ADC instruction.
6. Write an assembly language program write an assembly language program to subtract two 16-bit numbers.
7. Write an assembly language program to find the largest and smallest number in an array of data using 8085 instruction set.
8. Write an assembly language program to write a program to arrange an array of data in ascending and descending order.
9. Write an assembly language program to convert given Hexadecimal number into its equivalent ASCII number and vice versa using 8085 instruction set.
10. Write an assembly language program for displaying fixed message and rolling message using 8051 Microcontroller.

11. Write an assembly language program for keyboard display interface 8279 and to execute the following programs.
    II. To display character ‘A’
    III. To read a key
12. Write an assembly language program to check whether a number is prime or not.
13. Write an assembly language program find 1’s complement of a number.
14. Write an assembly language program to add two BCD numbers.
15. Write an assembly language program to find the factorial of a number.
16. Write an assembly language program to find first 20 Fibonacci number.

**Suggested Readings:**


1. Barry B. Brey : The Intel Microprocessors : Architecture, Programming and Interfacing. Pearson Education, Sixth Edition, 2009.
2. Walter A Triebel, Avtar Singh; The 8085 and 8086 Microprocessors Programming, Interfacing, Software, Hardware, and Applications. PHI, Fourth Edition 2005.

## *MAJOR ELECTIVE (DSE)*

**Major Elective -2: Internet of thing (IoT)** **Credits 04(Full Marks: 75)**

**MJ DSE-2T: Internet of thing (IoT) (Theory)** **Credits 03**

**Course contents:**

**UNIT-I**

Fundamentals of IoT: Introduction, Definitions & Characteristics of IoT, IoT Architectures, Physical & Logical Design of IoT, Enabling Technologies in IoT, History of IoT, About Things in IoT, The Identifiers in IoT, About the Internet in IoT, IoT frameworks, IoT and M2M.

**UNIT-II**

Sensors Networks : Definition, Types of Sensors, Types of Actuators, Examples and Working, IoT Development Boards: Arduino IDE and Board Types, RaspberriPi Development Kit, RFID Principles and components, Wireless Sensor Networks: History and Context, The node, Connecting nodes, Networking Nodes, WSN and IoT.

**UNIT- III**

**Data Handling& Analytics:** Introduction, Bigdata, Types of data, Characteristics of Big data, Data handling Technologies, Flow of data, Data acquisition, Data Storage, Introduction to Hadoop. Introduction to data Analytics, Types of Data analytics, Local Analytics, Cloud analytics and applications

**UNIT-IV**

**Applications of IoT:** Home Automation, Smart Cities, Energy, Retail Management, Logistics, Agriculture, Health and Lifestyle, Industrial IoT, Legal challenges, IoT design Ethics, IoT in Environmental Protection.

**Text Books:**


1. Hakima Chaouchi, ― "The Internet of Things Connecting Objects to the Web" ISBN : 978-1- 84821-140-7, Wiley Publications
2. Olivier Hersent, David Boswarthick, and Omar Elloumi, ― "The Internet of Things: Key Applications and Protocols", WileyPublications
3. Vijay Madisetti and ArshdeepBahga, ― "Internet of Things (A Hands-on-Approach)", 1st Edition, VPT, 2014.

4. J. Biron and J. Follett, "Foundational Elements of an IoT Solution", O'Reilly Media, 2016.
5. Keysight Technologies, “The Internet of Things: Enabling Technologies and Solutions for Design and Test”, Application Note, 2016.

**References**

6. Daniel Minoli, ― “Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications”, ISBN: 978-1-118-47347-4, Willy Publications
7. Pethuru Raj and Anupama C. Raman, "The Internet of Things: Enabling Technologies, Platforms, and Use Cases", CRC Press
8. https://onlinecourses.nptel.ac.in/noc17_cs22/course
9. http://www.cse.wustl.edu/~jain/cse570-15/ftp/iot_prot/index.html

**MJ DSE-2P: Practical of IoT (Arduino Lab)** **Credits 01**

1. Understanding Arduino UNO Board and Components
2. Installing and work with Arduino IDE
3. Blinking LED sketch with Arduino
4. Simulation of 4-Way Traffic Light with Arduino
5. Ultrasonic sensor handing with Arduino
6. Servo motor, stepper handing with Arduino
7. Different sensor handing related to smart home using Arduino.
8. Spinning a DC Motor and Motor Speed Control Sketch
9. Working with Shields
10. Interfacing Arduino with Cloud.

**OR**

**Major Elective -2 Block-Chain** **Credits 04(Full Marks: 75)**

**MJ DSE-2T: Block-Chain (Theory)** **Credits 04**

**Course contents:**

**Introduction:** **8Hrs.**

Need for Distributed Record Keeping, Modeling faults and adversaries, Byzantine Generals problem. Consensus algorithms and their scalability problems Why Nakamoto Came up with Blockchain based cryptocurrency? Technologies Borrowed in Blockchain – hash pointers, consensus, byzantine fault-tolerant distributed computing, digital cash etc.

**Basic Distributed Computing:** **8Hrs.**

Atomic Broadcast, Consensus, Byzantine Models of fault tolerance

**Basic Crypto primitives:** **8Hrs.**

Hash functions, Puzzle friendly Hash, Collison resistant hash, digital signatures, public key crypto, verifiable random functions, Zero-knowledge systems

**Blockchain 1.0 :** **8Hrs.**

Bitcoin blockchain, the challenges, and solutions, proof of work, Proof of stake, alternatives to Bitcoin consensus, Bitcoin scripting language and their use

**Blockchain 2.0 :** **10Hrs.**

Ethereum and Smart Contracts, The Turing Completeness of Smart Contract Languages and verification challenges, Using smart contracts to enforce legal contracts, comparing Bitcoin scripting vs. Ethereum Smart Contracts

**Blockchain 3.0**: **8Hrs.**

Hyperledger fabric, the plug and play platform and mechanisms in permissioned blockchain

**Privacy, Security issues in Blockchain :** **10Hrs.**

Pseudo-anonymity vs. anonymity, Zcash and Zk-SNARKS for anonymity preservation, attacks on Blockchains – such as Sybil attacks, selfish mining, 51% attacks - -advent of algorand, and Sharding based consensus algorithms to prevent these

**Reference Books:**

1. Josh Thompson, 'Blockchain: The Blockchain for Beginnings, Guild to Blockchain Technology and Blockchain Programming', Create Space Independent Publishing Platform, 2017.

**OR**

**Major Elective -2 Cloud Computing** **Credits 04(Full Marks: 75)**

**MJ DSE-2T: Cloud Computing (Theory)** **Credits 03**

**Course contents:**

**Overview of Computing Paradigm:** **8Hrs.**

Recent trends in Computing: Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, Cloud Computing,

**Introduction to Cloud Computing** **8Hrs**

Introduction to Cloud Computing, History of Cloud Computing, Cloud service providers, Benefits and limitations of Cloud Computing,

**Cloud Computing Architecture** **12 Hrs**

Comparison with traditional computing architecture (client/server), Services provided at various levels, Service Models- Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), How Cloud Computing Works, Deployment Models- Public cloud, Private cloud, Hybrid cloud, Community cloud, Case study of NIST architecture.

**Case Studies** **8Hrs**

Case study of Service model using Google App Engine, Microsoft Azure, Amazon EC2, Eucalyptus.

**Service Management in Cloud computing** **12 Hrs**

Service Level Agreements (SLAs), Billing & Accounting, Comparing Scaling Hardware: Traditional vs. Cloud, Economics of scaling.

**Cloud Security** **12 Hrs**

Infrastructure Security- Network level security, Host level security, Application level security, Data security and Storage- Data privacy and security Issues, Jurisdictional issues raised by Data location, Authentication in cloud computing.

**Reference Books:**


1. Cloud Computing: Principles and Paradigms, Editors: Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Wile, 2011
2. Cloud Computing: Principles, Systems and Applications, Editors: Nikos Antonopoulos, Lee Gillam, Springer, 2012
3. Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Ronald L. Krutz, Russell Dean Vines, Wiley-India, 2010

4. Gautam Shroff, Enterprise Cloud Computing Technology Architecture Applications , Adobe Reader ebooks available from e-Books.com,2010
5. Toby Velte, Anthony Velte, Robert Elsenpeter, Cloud Computing, A Practical Approach ,McGraw Hills, 2010.
6. Dimitris N. Chorafas, Cloud Computing Strategies , CRC Press, 2010

**MJ DSE-2P: Cloud Computing Lab (Practical) Credits 01**

1. Create virtual machines that access different programs on same platform.
2. Create virtual machines that access different programs on different platforms.
3. Working on tools used in cloud computing online-
    a) Storage
    b) Sharing of data
    c) Manage your calendar, to-do lists,
    d) A document editing tool
4. Exploring Google cloud
5. Exploring microsoft cloud

**OR**

**Major Elective -2: Embedded System** **Credits 04(Full Marks: 75)**

**Objective:**

1. To have knowledge about the basic working of a microcontroller system and its programming in assembly language.
2. To provide experience to integrate hardware and software for microcontroller applications systems.

**Outcome:**

To acquire knowledge about microcontrollers embedded processors and their applications.

1. Foster ability to understand the internal architecture and interfacing of different peripheral devices with Microcontrollers.
2. Foster ability to write the programs for microcontroller.
3. Foster ability to understand the role of embedded systems in industry.
4. Foster ability to understand the design concept of embedded systems.

**MJ DSE-2T: Cloud Computing (Theory)** **Credits 03**

**Course contents:**

1. Introduction- Embedded system overview, embedded hardware units, embedded software in a system, embedded system on chip (SOC), design process, classification of embedded systems.
2. Embedded computing platform - CPU Bus, memory devices, component interfacing, networks for embedded systems, communication interfacings: RS232/UART, RS422/RS485, IEEE 488 bus.
3. Survey of software architecture- Round robin, round robin with interrupts, function queue scheduling architecture, selecting an architecture saving memory space.
4. Embedded software development tools- Host and target machines, linkers, locations for embedded software, getting embedded software into target system, debugging technique.
5. RTOS concepts - Architecture of the kernel, interrupt service routines, semaphores, message queues, pipes.
6. Instruction sets- Introduction, preliminaries, ARM processor, SHARC processor.
7. System design techniques - Design methodologies, requirement analysis, specifications, system analysis and architecture design.
8. Design examples- Telephone PBX, ink jet printer, water tank monitoring system, GPRS, Personal Digital Assistants, Set Top boxes

**References:**


1. Computers as a component: principles of embedded computing system design- wayne wolf
2. An embedded software premier: David E. Simon
3. Embedded / real time systems-KVKK Prasad, Dreamtech press, 2005
4. Programming for embedded system by Dr. Prasas, Vikas Gupta, Das & Verma, Pub, WILEY Dreamtech india Pvt.
5. Embadded System Design. by Frank Vashid & Tony Givergis, Pub, WILEY.
6. MFC Programming. by Herbert Schildt, Pub. TataMcGraw Hill.
7. Programming Embedded

## *MINOR (MI)*

**Minor (MI) -6: Database Management Systems** **Credits 04 (Full Marks: 75)**

**OBJECTIVE OF THE COURSE**

- Provide a comprehensive understanding of database design, implementation, and management.
- Cover fundamental concepts of database architecture, data models, and relational database management systems (RDBMS).
- Teach students to design and normalize databases to ensure data integrity and eliminate redundancy.
- Emphasize the use of SQL for querying and manipulating databases, including creating, updating, and deleting records.
- Explore advanced topics such as transactions, concurrency control, and database security for reliable data handling.
- Offer hands-on experience through projects involving the design and implementation of database systems.
- Equip students with skills to design, manage, and optimize databases.
- Prepare students for careers in database administration, data analysis, and software development.

**MI -6T: Database Management Systems (Theory)** **Credits 03**

**Course contents:**

**Module- I Introduction** **6 Hrs.**

Characteristics of database approach, data models, database system architecture and data independence

**Module- II Entity Relationship(ER) Modeling** **6 Hrs.**

Entity types, relationships, constraints.

**Module- III Relation data model** **10 Hrs.**

Relational model concepts, relational constraints normalization, relational algebra, SQL queries

**Module- IV Database design** **20 Hrs.**

Mapping ER/EER model to relational database, functional dependencies, Lossless decomposition, Normal forms (up to BCNF).

**Module- V Transaction Processing** **10 Hrs.**

ACID properties, concurrency control

**Module- VI File Structure and Indexing 8 Hrs.**

Operations on files, File of Unordered and ordered records, overview of File organizations, Indexing structures for files( Primary index, secondary index, clustering index), Multilevel indexing using B and $B^+$ trees.

**Recommended Books:**


1. R. Elmasri, S.B. Navathe, Fundamentals of Database Systems 6th Edition, Pearson Education, 2010.
2. R. Ramakrishanan, J. Gehrke, Database Management Systems 3rd Edition, McGraw-Hill, 2002.
3. A. Silberschatz, H.F. Korth, S. Sudarshan, DatabaseSystem Concepts 6th Edition, McGraw Hill, 2010.
4. R. Elmasri, S.B. Navathe Database Systems Models, Languages, Design and application Programming, 6th Edition, Pearson Education, 2013.


**MI -6P: Database Management Systems Lab (Practical) Credits 01**

**A. Create and use the following database schema to answer the given queries**

**EMPLOYEE Schema**

| Field | Type | NULL | KEY | DEFAULT |
|---|---|---|---|---|
| **Eno** | Char(3) | | PK | |
| Ename | Varchar(50) | | | |
| Job_type | Varchar(50) | | | Part |
| Manager | Char(3) | | | |
| Hire_date | Date | | | |
| Dno | Integer | | FK | |
| Commission | Decimal(10,2) | | | |
| Salary | Decimal(10,2) | | | |

**DEPARTMENT Schema**

| Field | Type | NULL | KEY | DEFAULT |
|---|---|---|---|---|
| Dno | Integer | | PK | |
| Dname | Varchar(50) | NOT NULL | | |
| Location | Varchar(50) | | | KOLKATA |

**Query List**

1. Query to display Employee Name, Job, Hire Date, Employee Number; for eachemployee with the Employee Number appearing first.

2. Query to display unique Jobs from the Employee Table.
3. Query to display the Employee Name concatenated by a Job separated by a comma.
4. Query to display all the data from the Employee Table. Separate each Column by a comma and name the said column as THE_OUTPUT.
5. Query to display the Employee Name and Salary of all the employees earning more than $2850.
6. Query to display Employee Name and Department Number for the Employee No= 7900.
7. Query to display Employee Name and Salary for all employees whose salary is not in the range of $1500 and $2850.
8. Query to display Employee Name and Department No. of all the employees in Dept 10 and Dept 30 in the alphabetical order by name.
9. Query to display Name and Hire Date of every Employee who was hired in 1981.
10. Query to display Name and Job of all employees who don‘t have a current Manager.
11. Query to display the Name, Salary and Commission for all the employees who earn commission.
12. Sort the data in descending order of Salary and Commission.
13. Query to display Name of all the employees where the third letter of their name is _A‘.
14. Query to display Name of all employees either have two _R‘s or have two _A‘s in their name and are either in Dept No = 30 or their Manger‘s Employee No = 7788.
15. Query to display Name, Salary and Commission for all employees whose Commission Amount is 14 greater than their Salary increased by 5%.
16. Query to display the Current Date.
17. Query to display Name, Hire Date and Salary Review Date which is the 1st Monday after six months of employment.
18. Query to display Name and calculate the number of months between today and the date each employee was hired.
19. Query to display the following for each employee <E-Name> earns < Salary> monthly but wants < 3 * Current Salary >. Label the Column as Dream Salary.
20. Query to display Name with the 1st letter capitalized and all other letter lower case and length of their name of all the employees whose name starts with _J‘, ‘A‘ and _M‘.

**B.** Create a database having two tables with the specified fields, to computerize a library system of a Delhi University College.

**Library Books (Accession number, Title, Author, Department, Purchase Date, Price)**
**Issued Books (Accession number, Borrower)**

h) Identify primary and foreign keys. Create the tables and insert at least 5 records in each table.
i) Delete the record of book titled ―Database System Concepts‖.

j) Change the Department of the book titled ―Discrete Maths‖ to ―CS‖.
k) List all books that belong to ―CS‖ department.
l) List all books that belong to ―CS‖ department and are written by author ―Navathe‖.
m) List all computer (Department=‖CS‖) that have been issued.

List all books which have a price less than 500 or purchased between ―01/01/1999‖ and ―01/01/2004‖