

1. Name of the depositor.
 2. Account number.
 3. Type of account.
 4. Balance amount in the account.
 - Member Functions
 5. To assign initial values.
 6. To deposit an amount.
 7. To withdraw an amount after checking the balance.
 8. To display name and balance.
 - Use appropriate main program. (application level calsspgm)
15. Create a base class called shape. Use this class to store two double type values that could be used to compute the area of figures. Derive two specific classes called TRIANGLE and RECTANGLE from the base SHAPE. Add to the base lass, a member function get_data() to initialize base class data members and another member function display_area() to compute and display the area of figures. Make display_area() as a virtual function and redefine this function in the derived class to suite the requirements (virtual functions)

GENERAL AWARENESS COURSE II : 3A12BCA DATA STRUCTURES

SEMESTER	COURSE CODE	HOURS PER WEEK	CREDIT	EXAM HRS
III	3A12BCA	4	4	3

COURSE OUTCOME

- CO1:** Understand the concept of data structures and its relevance in computer science.
CO2: Familiarize with selected linear and nonlinear data structures.
CO3: Enhance skill in programming.

Unit I:

Data structures: Definition and Classification. Array: - Operations; Number of elements; Array representation in memory. Polynomial representation with arrays; Polynomial addition. Sparse matrix: Addition of sparse matrices. The concept of recursion: types, example: factorial and Tower of Hanoi problem . (12 Hrs)

Unit II:

Sorting algorithms: Insertion, bubble, selection, quick and merge sort; Comparison of Sort algorithms. Searching techniques: Linear and Binary search. (12 Hrs)

Unit III:

Stack: Operations on stack; array representation. Application of stack- i. Postfix expression evaluation. ii. Conversion of infix to postfix expression. Queues: Operation on queue. Circular queue; Dequeue, and priority queue. Application of queue: Job scheduling. (15 Hrs)

Unit IV:

Linked list – Comparison with arrays; representation of linked list in memory. Singly linked list- structure and implementation; Operations – traversing; Add new node; Delete node; Reverse a list; Search and merge two singly linked lists. Stack with singly linked list. Circular linked list – advantage. Queue as Circular linked list. Doubly linked list – structure; Operations – Add/delete nodes; Advantages. (15 Hrs)

Unit V:

Tree and Binary tree: Basic terminologies and properties; Linked representation of Binary tree; Complete and full binary trees; Binary tree representation with array. Tree traversal: in order, pre order and post order traversals. Binary Search Tree. Application of binary tree: Huffman Code.

(10 Hrs)

Books for Study:

1. Classic Data structures, Samanta, Second Edition, PHI

Books for Reference:

1. Data Structures and Algorithms: Concepts, Techniques and Applications; GAV Pai, Mc Graw Hill, 2008
2. Fundamentals of Data structures in C++ , 2nd Edn, Horowitz Sahni, Anderson, Universities Press

Books for Study:

1. Discrete Mathematics and Its Applications with Combinatorics and Graph Theory, Kamala Krithivasan, McGraw Hill Education, 2011 (Seventh Edition).

Books for Reference:

1. J. K. Sharma, *Discrete Mathematics*, Macmillan Publishers India Limited, ISBN: 1403924759.
2. Alan Doerr and Kenneth Levassur, *Applied Discrete Structure for Computer Science*, Galgotia Publications Pvt. Ltd, ISBN: 9780574217554.
3. Discrete Mathematics , N Ch S N Iyengar, V M Chandrasekharan, KA Venkatesh, PS Arunachalam, Vikas Publishing , 2003.
4. C Liu and D. Mohapatra, *Elements of Discrete Mathematics - A Computer Oriented Approach*, TMH, ISBN: 1259006395.

**GENERAL AWARENESS COURSE V: 4A15BCA LAB -III
DATA STRUCTURES & DBMS**

SEMESTER	COURSE CODE	HOURS PER WEEK	CREDIT	EXAM HRS
IV	4A15BCA	III SEM 3 Hrs, IV SEM 2 Hrs	2	3

Program List**Section A: DATA STRUCTURE**

1. Add two polynomials.
2. Sequential and binary search : Print number of comparison in each case for given datasets.
3. Insertion sort: number of comparisons and exchanges for given data sets.
4. Bubble sort: Print number of comparisons and exchanges for given data sets.
5. Selection sort: Print number of comparisons and exchanges for given data sets .
6. Quick sort.
7. Stack operation: addition and deletion of elements

8. Queue operation: addition and deletion of elements
9. Conversion of infix expression to postfix.
10. Menu driven program: to add / delete elements to a circular queue. Include necessary error messages.
11. Singly linked list operations : add a new node at the beginning, at the end, after ith node, delete from beginning, end, print the list.
12. Circular linked list : add a new node at the beginning, at the end, after ith node, delete from beginning, end, print the list.
13. Doubly linked list : add a new node at the beginning, at the end, after ith node, delete from beginning, end, print the list.
14. Implement tree traversal.
15. Merge two sorted linked list.

Section B: DBMS

Minimum 10 exercises covering SQL related topics. Sample exercises are given below:

SQL -1

Create table students with fields sno, sname, sex, mark with sno as primary key and assign suitable constraints for each attribute. Insert five records into the table.

1. Alter the table by adding one more field rank.
2. Display all boy students with their name.
3. Find the Average mark
4. Create a query to display the sno and sname for all students who got More than the
5. average mark. Sorts the results in descending order of mark.
6. Display all girl student names for those who have marks greater than 20 and less than 40.

SQL -2

Create a table department with fields ename, salary, dno, dname, place with dno as primary key. Insert five records into the table.

1. Rename the field 'place' with 'city'
2. Display the employees who got salary more than Rs.6000 and less than 10000 /-
3. Display total salary of the organization
4. Display ename for those who are getting salary in between 5000 and 10000.
5. Create a view named 'Star' with field ename, salary & place
6. Display ename and salary with salary rounded with 10 digits**'

SQL -3

Create a table department with fields dno, dname, dmanager and place with dno as primary key.

Create a table emp with fields eno, ename, job, dno, salary, with eno as primary key. Set dno as foreign key.