



- **KEEPING MOBILE PHONE/ANY ELECTRONIC GADGETS, EVEN IN 'OFF' POSITION IS TREATED AS EXAM MALPRACTICE**
- **DON'T WRITE ANYTHING ON THE QUESTION PAPER**

Answer ALL Questions

(10 X 10 = 100 Marks)

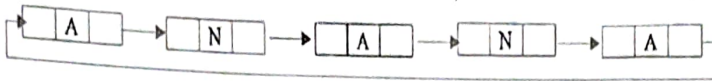
1. You are managing a popular theme park ride with a fixed number of seats per ride cycle (ex: 10 seats). Visitors form a queue to get on the ride, and you need an efficient and optimized way to manage this queue, ensuring that the ride operates at full capacity and minimize wait times. Describe how you would set up to manage visitors waiting for the ride. Implement functions to add and remove visitors from the queue. Explain how the solution ensures that the ride operates at full capacity and minimizes visitor wait times. Write a Pseudocode to implement insertion, deletion and display the elements.
- 2.(a) ✎ You are developing a text editor that allows users to write, edit, and format text. One of the key features is the ability to undo and redo changes, ensuring that users can easily revert mistakes and redo actions if necessary. Identify the data structure suitable for the given application and explain the data structure with suitable operations.

OR

- 2.(b) Let S1 and S2 be two stacks of integers. The size of S1 and S2 are n and m respectively. Assume z is an integer. A Set SUM is defined as follows.
SUM = { x ≤ z : x = y1 + y2, where y1 ∈ S1 and y2 ∈ S2, S1.top = y1 and S2.top = y2 }
That is, the set SUM have integers from two stacks having sum less than or equal to z.
Write a Pseudocode to compute the total number of elements in SUM using push and pop operations. For example z = 100, S1 = { 50, 88, 25, 70}. and S2 = { 100, 10, 11, 250}, 88+10 < 100 and 25+11 < 100. Output = 2.
3. Write a program / pseudocode to create a circularly linked list (insertion and display) by storing each node with a character. Read the characters stored in the linked list in rotated manner and check the collection of characters form palindrome.

Traverse the nodes like first time start from first node to last node. Second time from second node to first node. Third time third node to second node and so on

Ex: Linked list



Output:

ANANA is palindrome

NANAA is not palindrome

ANAAN is not palindrome

NAANA is not palindrome

AANAN is not palindrome

4.

Given the following code segments, Find the time complexity for each and represent the time complexity in ascending order. Illustrate the growth rate using graphical representation.

a) void func1(int n)

```

{
    int i, j;
    for(i=1; i <= n/3; i++)
    for(j=1; j <= n; j += 4)
    cout<<i<<j;
}
  
```

b) void func2()

```

{
    int i;
    int n;
    cin>>n;
    for(i=1; i*i <= n; i++)
    cout << i << endl;
}
  
```

c) void func3()

```

{
    int i,j,k;
    for(i=n/2; i <= n; i++)
    for(j=1; j <= n/2 ; j++)
    for(k=1; k <= n; k *= 2)
    cout << i<<j<<k<<endl;
}
  
```

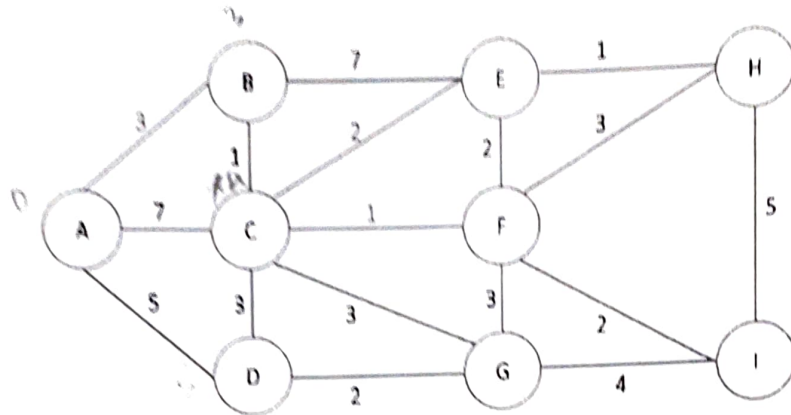
5. Assume you are given an array of integers representing the ages of people in a group: [25, 32, 40, 19, 60, 55, 22]. You need to sort this array in ascending order using heap sort with a min heap. Additionally, after sorting the array, two new people join the group with ages 35 and 18. You need to insert these two new ages into the sorted array while maintaining the sorted order.
- i) How would you perform heap sort on the given array using a min heap? [5]
- ii) How would you insert the two new ages into the sorted array while maintaining the sorted order? [5]

6. You are tasked with developing an employee management system for a company that needs to generate various reports on its employees. Each employee record contains attributes such as `employee_id`, `name`, `department`, `salary`, and `hire_date`. The system requires efficient storage, retrieval, and manipulation of employee records for reporting purposes. To facilitate this, you decide to implement a Binary Search to search the employee details using `employee_id`. Implement Binary Search using linked list.

7. Construct a binary search tree for the given strings: judy mary , bill, alice, tom, fred, jane and joe. Write an algorithm to insert nodes in a tree and perform the following operations:
- Find the in-order, pre-order, post-order, and level order traversal
 - Delete alice, mary, judy and fred
 - Find the in-order, pre-order, post-order, and level order traversal after deletion.

8. You are working on optimizing the indexing mechanism for a high-performance database system. Given the need to manage large datasets efficiently, you have decided to implement B-trees for indexing. B-trees are self-balancing tree data structures that maintain sorted data and allow searches, sequential access, insertions, and deletions in logarithmic time. Describe the step-by-step process of inserting and deleting elements in a B-tree. Highlight how B-trees ensure balance and minimize disk read/write operations using suitable example.

9. For the given weighted graph $G=(V,E)$.



- Output the sequence of vertices identified by the Dijkstra's algorithm for single source shortest path when the algorithm is started at node A. [4]
 - What is the cost of shortest path from A to F? [2]
 - Write the Dijkstra's algorithm / pseudocode and demonstrate the process step by step. [4]
- 10.(a) Illustrate the working of quicksort on the array $A = [9, 7, 5, 11, 12, 2, 14, 3, 10, 6]$. Assume that the first element is chosen as the pivot when the algorithm partitions the array. Show the array after every partition and trace the recursive calls made by the algorithm.

OR

- 10.(b) A software company has developed a new text-based application that handles large volumes of data. To optimize storage space and improve transmission speeds, they are interested in implementing Huffman coding for data compression. The application frequently processes various documents containing characters with differing frequencies. Discuss how Huffman Coding efficiently handles large files. Demonstrate step by step process for constructing a Huffman tree for the given input.

character	Frequency
a	5
b	9
c	12
d	13
e	16
f	45

==== J/L/TX =====