Q1. What do you mean by a Data structure?
Ans : Data Structure is the data organization, management and storage format that enables efficient access and modification of data.
There are two types of data structures :-
i) Linear Data Structure – Array, LinkedList, Stack, Queue
ii) Non Linear Data Structure – Tree, Graphs

Q2. What are some of the applications of DS?
Ans : Applications of DS are : -
i) Array
ii) LinkedList
iii) Stack
iv) Queue
v) Tree
vi) Graph, etc

Q3. What are the advantages of a Linked list over an array?

Arrays	LinkedList
Size is fixed.	Size is not fixed.
Memory is allocated from Stack.	Memory is allocated from Heap.
Necessary to specify no of elements during declaration.	No Necessary to specify no of elements during declaration.
Deleting an element is not possible.	Deleting an element is possible.
Inserting an element at front is difficult as we have to shift the element first.	Inserting an element at any position can be carried out easily.

Q4. What are the minimum number of Queues needed to implement the priority queue?
Ans : Two.

Q5. Write the syntax in C to create a node in the singly linked list.

```
 #include <stdio.h>
 #include <stdlib.h>
 struct node{
 int data;
 struct node *next;
};
```

Q6. What is the use of a doubly-linked list when compared to that of a singly linked list?

Ans : Singly linked list allows traversal elements only in one way. Doubly linked list allows element two way traversal. On other hand doubly linked list can be used to implement stacks as well as heaps and binary trees

Q7. What is the difference between an Array and Stack?

Ans :

Stack	Array
Stacks are based on the LIFO principle, i.e., the element inserted at the last, is the first element to come out of the list.	In the array the elements belong to indexes, i.e., if you want to get into the fourth element you have to write the variable name with its index or location within the square bracket eg arr[4]
Insertion and deletion in stacks takes place only from one end of the list called the top.	Insertion and deletion in array can be done at any index in the array.
Stack has a dynamic size.	Array has a fixed size.
Stack can contain elements of different data type.	Array contains elements of same data type.
We can do only linear search	We can do both linear and Binary search

Q8. What are the different types of traversal techniques in a tree?

Ans: There are four types of traversal techniques in a tree :-

i) Inorder
ii) Pre Order
iii) Post Order
iv) Level Order

Q9. Why it is said that searching a node in a binary search tree is efficient than that of

a simple binary tree?

Ans : It supports three main operations: searching of elements, insertion of elements, and deletion of elements. Binary Search Tree allows for fast retrieval of elements stored in the tree as each node key is thoroughly compared with the root node, which discards half of the tree.

Q10. What are the applications of Graph DS?

Ans : Graph is used to represent networks of communication, data organization, computational devices etc.

Q11. Can we apply Binary search algorithm to a sorted Linked list?

Ans : Yes

Q12.When can you tell that a Memory Leak will occur?

Ans: A memory leak is a type of resource leak that occurs when a computer program incorrectly manages memory allocations in a way that memory which is no longer needed is not released. A memory leak may also happen when an object is stored in memory but cannot be accessed by the running code.

Q13. How will you check if a given Binary Tree is a Binary Search Tree or not?

Ans: A Binary Search Tree (BST) is a binary tree with the following properties:

i) The left subtree of a particular node will always contain nodes whose keys are less than that node's key.
ii) The right subtree of a particular node will always contain nodes with keys greater than that node's key.
iii) The left and right subtree of a particular node will also, in turn, be binary search trees.

Q14. Which data structure is ideal to perform recursion operation and why?

Ans: Stack. Because of its LIFO (Last In First Out) property it remembers its 'caller' so knows whom to return when the function has to return. Recursion makes use of system stack for storing the return addresses of the function calls. Every recursive function has its equivalent iterative (non-recursive) function.

Q15. What are some of the most important applications of a Stack?

Ans : Applications of Stack are :

i) Stacks can be used for expression evaluation.
ii) Stacks can be used to check parenthesis matching in an expression.
iii) Stacks can be used for Conversion from one form of expression to another.
iv) Stacks can be used for Memory Management.
v) Stack data structures are used in backtracking problems.

Q16. Convert the below given expression to its equivalent Prefix And Postfix notations.

Ans : Incomplete Ques……………..