Q1.What do you mean by data structures?

A. A data structure is a particular way of organizing the data in a computer so that we can utilize it more effectively.

Q2. What are the some applications of data structures?

A. Some basic applications of data structures are:

1. Search
2. Sort
3. Insert
4. Delete
5. Update

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

A. There are two advantages of a linked list over an array:

- The size of an array is fixed whereas the size of a linked list is dynamic
- The insertion and deletion in linked list is easier because the elements of an array has to move during these operations'

Q4. Write a syntax in C to create a node of a single linked list

A.
```
struct node{
    int head;
   node *next;
};
int main(){
node first=new node();
first=(node)(malloc(sizeof(struct node)));
first->head=1;
first->next=null;
}
```

Q18. Program to reverse a queue

A.
```
#include<iostream>
#include<queue>
#include<stack>
using namespace std;
void rq(queue<int>& Queue)
```

```
{
	if(Queue.empty()==true)
	{
		return;
	}
   stack<int> Stack;
   for(int i=0;i<sizeof(Queue);i++)
   {
	Stack.push(Queue.front());
	Queue.pop();
	}
	while(!Stack.empty())
	{
		Queue.push(Stack.top());
		Stack.pop();
	}

}
void Print(queue<int> & Queue)
{
	while(!Queue.empty())
	{
		cout<<Queue.front()<<" ";
		Queue.pop();
	}
}
int main()
{
	queue<int> Q;
	int k=3;
```

```
    Q.push(10);
    Q.push(20);
    Q.push(30);
    Q.push(40);
    Q.push(50);
    Q.push(60);Q.push(70);
    Q.push(80);
    Q.push(90);
    Q.push(100);
    rq(Q);
    Print(Q);
}
```

Q19. Program to reverse first K elements of a queue

A

Q6. What are the minimum number of queues needed to implement the priority queue?

A. Two queues are required to implement the priority queue

Q5. What is the use of doubly linked list when compared to singly linked list?

A. In a doubly linked list insertion and deletion has a complexity o(1) which makes it more efficient than singly linked list.

In a doubly linked list we can implement two-way traversal.

In a doubly linked list we can implement stack, heap, tree whereas in a singly linked list we can only implement only stacks

Q7. What is the difference between stack and an array?

A. A stack follows LIFO principle whereas an array the elements are in indices.

In a stack insertion and deletion occurs only at the top whereas we can perform insertion and deletion on any index in an array.

A stack is dynamic in size whereas an array is static in size.

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

A. Inorder, preorder, postorder.

Q9. Why it is said that searching in a binary search tree is easier than binary tree?

A. Binary tree is unordered which makes the insertion, searching and searching slower whereas in a binary search tree the smaller elements are on left side and the greater elements are on right side which makes insertion, searching and deletion easier.

Q10. What are the applications of Graph DS?

A. We use Graphs to find the flow of computation and the shortest path in operating system problem.

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

A. Yes, we can apply Binary search algorithm to a sorted linked list.

Q12. When can you tell a memory leak has occurred?

A. when memory runs out even though the total amount of memory allocated does not exceed the total available.

Q13. How can you check if a given binary tree is a binary search tree or not?

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

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

Q14. Which datastrucure is ideal to implement recursion and why?

A. Stack. Because of its LIFO 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.

Q15. What are some of the most important applications of stack?

- Expression conversion
- Syntax parsing
- Backtracking
- Function Calls
- String reversal