Q1: What do you mean by a Data Structure?

Ans: A Data structure is a programming construct used to model real life data and stores the data

efficiently in the system’s memory so that we can execute instructions on it and get meaningful results.

Basic data structures are arrays, lists, maps etc.

Q2. What are some of the applications of DS?

Ans: Data structure are used to store data in different- different scenario we need different kind of

data structure some of the application of data structure is:-

1. Arrays: Implementation of other data structures, Execution of matrices and vectors,

Dynamic memory allocation, Pointer container, Control tables.

2. Stack: Evaluation of expressions, Backtracking, Runtime memory management,

Arrangement of books in a library.

3. Queue: Here, the data sent need not be received at the same rate at which it was sent. A

certain system resource is to be shared between different processes.

4. Linked-List: Representation of sparse matrices, Non-contiguous data storage,

Implementation of non-binary tree or other data structures, Dynamic memory

management, Equalizing parenthesis, Symbol tables.

5. Set: Mapping of data, Common data storage.

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

Ans: a) The Size of LinkedList is incremented at runtime. But in Array it is not possible.

b) The number of elements in the linked list are limited to the available memory space

while the number of elements in the array is limited to the size of an array.

c) LinkedList stores the elements dynamically while array stores statically.

Q4. Write the syntax in C to create a node in the singly linked list?

Ans: struct node

{

int data;

struct node \*next;

};

struct node \*head, \*ptr;

ptr = (struct node \*)malloc(sizeof(struct node));

Q5. What is the use of a doubly-linked list when compared to that of a singly

Linked list?

Ans: A doubly linked list can be traversed bidirectionally therefore it saves the overhead of traversal in

cases when we roughly know the place of an element.

Q6. What is the difference between an Array and Stack?

Ans. The main difference between array and stack is accessing of data and traversing of data in Array

we can access data by index only or traverse the array from the index where we want but in case of

stack we can only access the data in last in first out fashion and traversing is also done from last input

to first input only.

Q7. What are the minimum number of Queues needed to implement the priority

queue?

Ans: Two queue will be required to make a priority queue. One queue is used for actual storing of data and

another for storing priorities.

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

Ans: There are three type of traversal technique of tree.

a) Pre Order

b) In Order

c) Post Order

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

tree.

Ans: Searching a node in binary search tree is efficient because of property of binary search tree that

each left side node is lesser than root node and each right side node is greater than root node so

because of this searching will always follow a particular single path there is no need to traverse each

node like binary tree.

Q10: What are the applications of Graph DS?

Ans: Graph DS is used to model scenarios where a component has relations with multiple components.

Graph DS is used for modelling networks such as social media networks etc.

Q11: Can we use binary search algorithm to a sorted linked list?

Ans : We cannot apply binary search algorithm to a sorted linked list because the binary search heavily

relies on indexes , and the overhead to bootstrap an index system defeats the purpose.

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

Ans: GNU malloc. Under Linux using GNU libc, the kernel and/or C run-time will sometimes detect memory

allocation or usage errors without doing anything special in your code or using any external tool.

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

Ans. 1. If a node is a left child, then its key and the keys of the nodes in its right subtree are less than

its parent’s key.

2. If a node is a right child, then its key and the keys of the nodes in its left subtree are greater than

its parent’s key

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

Ans: Stacks are used to perform recursion as function on the top of the stack can executed , the function

can be popped and the data can be passed to the next function, this cycle continues until the required

data is passed to the original function.

Q15: Most important applications of Stack?

Ans : Stacks are used to implement recursion, can be used to balance parenthesis and is used to

implement calculations following BODMAS rule .

Q16: Convert the below given expression to its equivalent prefix and post notations

Ans: Question is Incomplete.

Q23: Check if a given graph is a tree or not?

Ans : To check if a given graph is a tree or not, we have to check if the graph contains a cycle. A a

graph is a tree if it doesn’t contain a cycle. Since it is not mentioned if it is a directed or un-directed

graph, a particular cycle detection algorithm can’t be prescribed but techniques used to detect cycle are

BFS or DFS cycle detection algorithm and Topological sorting.

Q25: How to find the shortest path between two vertices?

Ans: Shortest path between two vertices is determined by using a shortest path algorithm called

Dijkstra’s Algorithm. Dijkstra’s Algorithm is also called single source shortest path algorithm.