C++ Data Structures Interview Questions and Answers

1. What is a Data Structure?

A data structure is a way of organizing and storing data so that it can be accessed and modified efficiently.

2. Difference between Array and Linked List:

- Array: Contiguous memory, O(1) random access, costly insert/delete, fixed size.

- Linked List: Non-contiguous memory, O(n) sequential access, easy insert/delete, dynamic size.

3. How is a Stack implemented in C++?

Using arrays, linked lists, or STL's std::stack.

Example (array-based stack):

```
class Stack {
    int top;
public:
    int a[MAX];
    Stack() { top = -1; }
    bool push(int x);
    int pop();
    bool isEmpty();
};
```

4. Difference between Struct and Class in C++:

- Struct members are public by default.

- Class members are private by default.

5. Advantages of Pointers in Data Structures:

- Dynamic memory management

- Easier dynamic structures like trees, graphs

6. How is a Linked List reversed?

Traverse and reverse the next pointers.

7. What is a Binary Search Tree (BST)?

A tree where left child < parent < right child. Fast search, insert, delete.

8. Difference between std::map and std::unordered_map:

- std::map: Red-Black Tree, ordered keys, O(log n) operations.

- std::unordered_map: Hash table, unordered keys, O(1) average operations.

9. Memory Management for Linked Lists:

- Use 'new' to allocate nodes.

- Use 'delete' to free memory and avoid leaks.

10. What are Smart Pointers?

- Manage memory automatically.

- std::unique_ptr, std::shared_ptr avoid leaks.