PRACTICAL 11

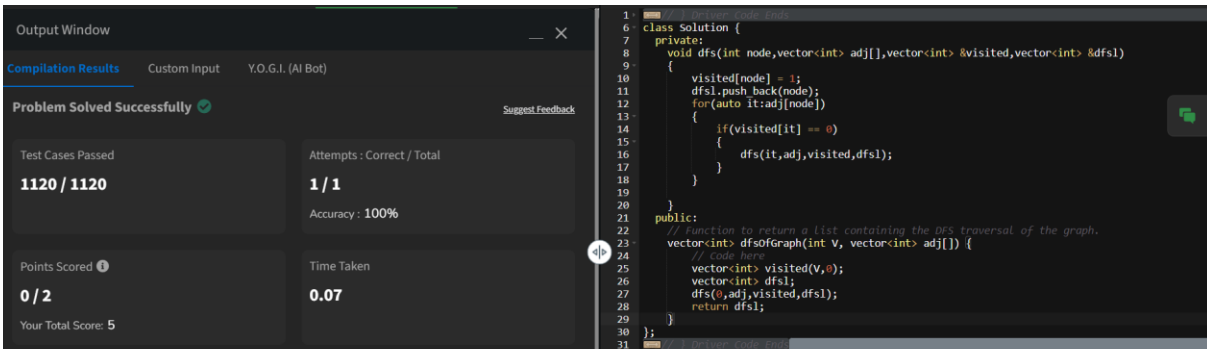
11.1 Implement DFS of Graph.
https://practice.geeksforgeeks.org/problems/depth-first-traversal-for-a-graph/1

CODE :

```cpp
class Solution {
  private:
    void dfs(int node,vector<int> adj[],vector<int> &visited,vector<int>
&dfsl)
    {
        visited[node] = 1;
        dfsl.push_back(node);
        for(auto it:adj[node])
        {
            if(visited[it] == 0)
            {
                dfs(it,adj,visited,dfsl);
            }
        }

    }
  public:
    // Function to return a list containing the DFS traversal of the graph.
    vector<int> dfsOfGraph(int V, vector<int> adj[]) {
        // Code here
        vector<int> visited(V,0);
        vector<int> dfsl;
        dfs(0,adj,visited,dfsl);
        return dfsl;
    }
};
```

OUTPUT:



Problem Solved Successfully

Test Cases Passed
1120 / 1120

Attempts : Correct / Total
1 / 1
Accuracy : 100%

Points Scored
0 / 2
Your Total Score: 5

Time Taken
0.07

```cpp
// } Driver Code Ends
class Solution {
  private:
    void dfs(int node,vector<int> adj[],vector<int> &visited,vector<int> &dfsl)
    {
        visited[node] = 1;
        dfsl.push_back(node);
        for(auto it:adj[node])
        {
            if(visited[it] == 0)
            {
                dfs(it,adj,visited,dfsl);
            }
        }

    }
  public:
    // Function to return a list containing the DFS traversal of the graph.
    vector<int> dfsOfGraph(int V, vector<int> adj[]) {
        // Code here
        vector<int> visited(V,0);
        vector<int> dfsl;
        dfs(0,adj,visited,dfsl);
        return dfsl;
    }
};
// } Driver Code Ends
```

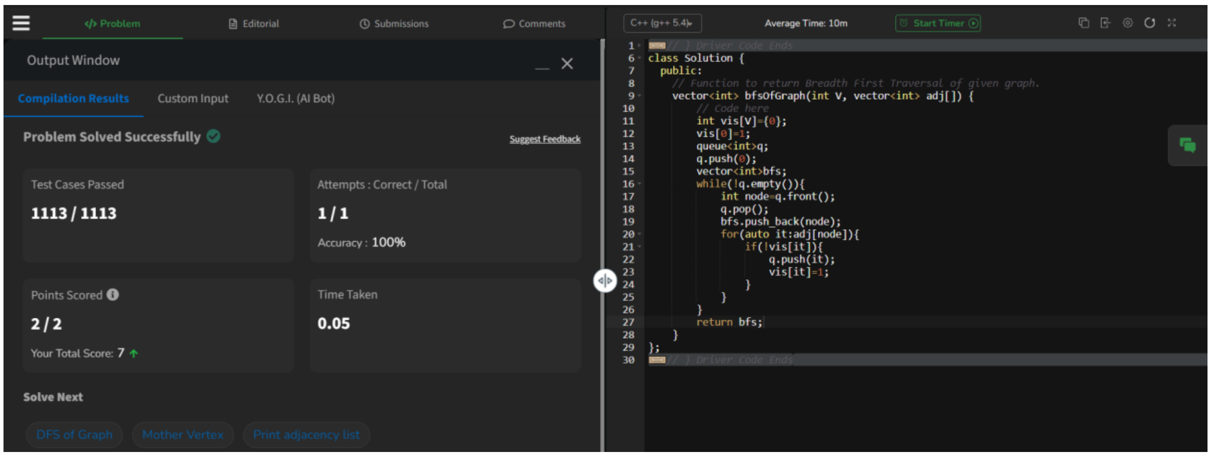
11.2 Implement BFS of Graph.
https://practice.geeksforgeeks.org/problems/bfs-traversal-of-graph/1

CODE:

```
class Solution {
  public:
    // Function to return Breadth First Traversal of given graph.
    vector<int> bfsOfGraph(int V, vector<int> adj[]) {
        // Code here
        //int n=adj.size();
        int vis[V]={0};
        vis[0]=1;
        queue<int>q;
        q.push(0);
        vector<int>bfs;
        while(!q.empty()){
            int node=q.front();
            q.pop();
            bfs.push_back(node);
            for(auto it:adj[node]){
                if(!vis[it]){
                    q.push(it);
                    vis[it]=1;
                }
            }
        }
        return bfs;

    }
};
```

OUTPUT:



Problem Solved Successfully

Test Cases Passed
1113 / 1113

Attempts : Correct / Total
1 / 1
Accuracy : 100%

Points Scored
2 / 2
Your Total Score: 7

Time Taken
0.05

Solve Next
DFS of Graph Mother Vertex Print adjacency list

```
// } Driver Code Ends
class Solution {
  public:
    // Function to return Breadth First Traversal of given graph.
    vector<int> bfsOfGraph(int V, vector<int> adj[]) {
        // Code here
        int vis[V]={0};
        vis[0]=1;
        queue<int>q;
        q.push(0);
        vector<int>bfs;
        while(!q.empty()){
            int node=q.front();
            q.pop();
            bfs.push_back(node);
            for(auto it:adj[node]){
                if(!vis[it]){
                    q.push(it);
                    vis[it]=1;
                }
            }
        }
        return bfs;
    }
};
// } Driver Code Ends
```