

* ***Name*:**
* **Aqeel Abbas**
* ***Roll Number*:**
* **56**
* ***University*:**
* **Superior University**
* ***Submitted to***
* **: sir Rasikh sahab**

***Question no 1***

***.*** *Dfs.*

#include <iostream>

#include <vector>

using namespace std;

struct TreeNode {

int data;

vector<TreeNode\*> children;

TreeNode(int val) : data(val) {}

};

void dfsTree(TreeNode\* node) {

if (!node) return;

cout << node->data << " ";

for (TreeNode\* child : node->children)

dfsTree(child);

}

int main() {

TreeNode\* root = new TreeNode(1);

root->children.push\_back(new TreeNode(2));

root->children.push\_back(new TreeNode(3));

root->children[0]->children.push\_back(new TreeNode(4));

root->children[0]->children.push\_back(new TreeNode(5));

root->children[1]->children.push\_back(new TreeNode(6));

cout << "DFS in Tree: ";

dfsTree(root);

return 0;

}

Dfs in graph

#include <iostream>

#include <vector>

using namespace std;

struct TreeNode {

int data;

vector<TreeNode\*> children;

TreeNode(int val) : data(val) {}

};

void dfsTree(TreeNode\* node) {

if (!node) return;

cout << node->data << " ";

for (TreeNode\* child : node->children)

dfsTree(child);

}

int main() {

TreeNode\* root = new TreeNode(1);

root->children.push\_back(new TreeNode(2));

root->children.push\_back(new TreeNode(3));

root->children[0]->children.push\_back(new TreeNode(4));

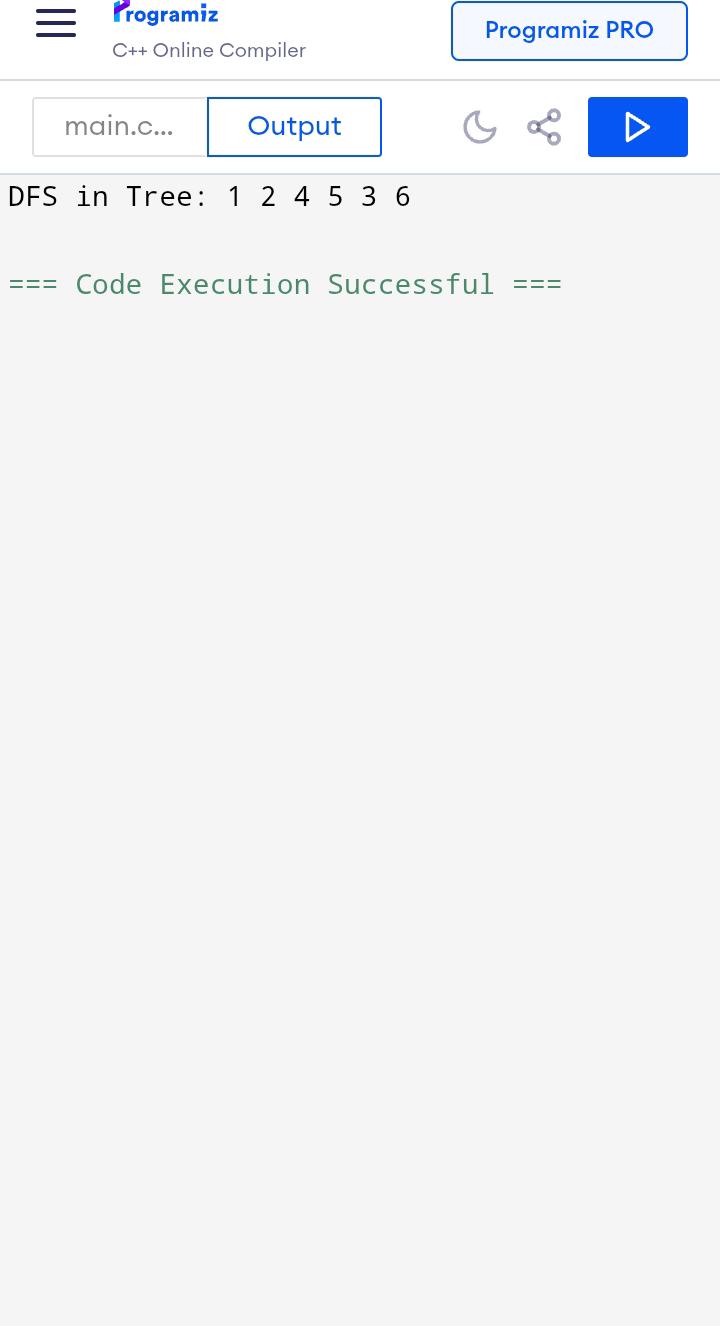
root->children[0]->children.push\_back(new TreeNode(5));

root->children[1]->children.push\_back(new TreeNode(6));

cout << "DFS in Tree: ";

dfsTree(root);

return 0;

}