G. H. Raisoni College Of Engineering And Management, Wagholi Pune

2021- 2022

Assignment no :- 7

Department	CE [SUMMER 2022 (Online)]		
Term / Section	III/B	Date Of **submission**	13-12-2021
Subject Name /Code	Data Structures and Algorithms/ UCSL201/UCSP201		
Roll No.	SCOB77	Name	Pratham Rajkumar pitty
Registration Number	2020AC0E1100107		

Experiment No 7

Aim: Implement binary tree using linked list and perform recursive traversals

Theory:

Trees represents the nodes connected by edges also a class of graphs that is acyclic is termed as trees.

- **Binary tree:→**

A binary tree is made of nodes, where each node contains a "left" reference, a "right" reference and a data element.

The top most node in tree is called the root

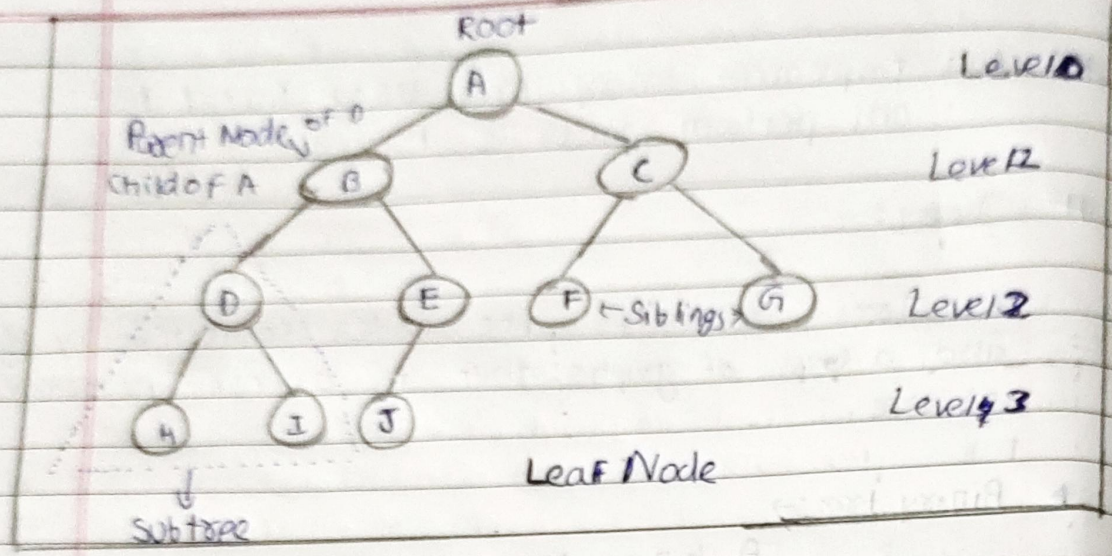
Each node can be connected to arbitrary number of nodes called children

Nodes with no children are called leaves or external nodes

Nodes which are not leaves are called internal nodes

Nodes with same parent are called Siblings

Every node (excluding a root) in a tree is connected by a directed edge from exactly one other node. This node is called the parent.



- Insert operation

The very first insertion creates the tree. Afterwards, whenever an element is to be inserted, first locate its proper location. Start searching from the root node, then if the data is less than the key value, search for the empty location in the left subtree and insert the data. Otherwise, search for the empty location in the right subtree and insert the data.

- Traversal → A traversal is the process to visit all nodes in a tree

There are 2 types of traversal

(1) depth-first traversal
(2) breadth-first traversal

- (1) depth-first traversal → 3 types in depth-first traversal

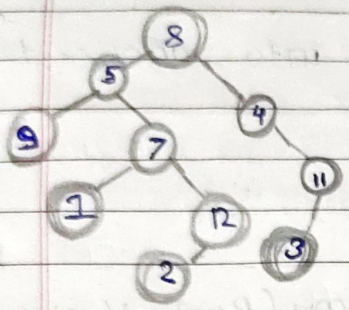
Preorder traversal - visit the parent first, then left, then right

Inorder traversal - Left, then parent, then Right child

Post order traversal - Left child, Right child, Then parent.

There is only one breadth first traversal
- The level order traversal

In this traversal only visits nodes by levels from top to bottom and from left to right.



Preorder - 8, 5, 9, 7, 1, 12, 2, 4, 11, 3
Inorder - 9, 5, 1, 7, 2, 12, 8, 4, 3, 11
PostOrder - 9, 1, 2, 12, 7, 5, 3, 11, 4, 8
LevelOrder - 8, 5, 4, 9, 7, 11, 1, 12, 3, 2

▶ Algorithm to insert a node:

Step1 - Search for the node whose child node is to be inserted. This is a node at some level i, and a node is to be inserted at the level i+1 as either its left child or right child. This is the node after which the insertion is to be made

Step2 :- Link a new node to the node that becomes its parent node, that is, either the L child or the R child.

Algorithm to traverse a tree

Inorder	Pre Post order
Until all nodes are transversed Step 1 - Recursively traverse left subtree. Step2 - Visit root node Step3 → Recursively traverse from right subtree.	Until all nodes are traversed Step1 :→ Visit root node Step2 :→ R.T. Left subtree Step3 :→ Recursively Traverse Right subtree.

Post order
Untill all nodes are traversed
Step1 :- Recursively traverse Left subtree.
Step2 :- Recursively traverse right subtree
Step3 :- Visit Root Node.

Algorithm to copy one tree into another tree.

Step1 : if (ROOT == NULL)
Then return NULL
Step2 : Temp = new TreeNode
Step3 : Temp -> LChild = Tree_copy (Root -> L child);
Step4 : Temp -> RChild = Tree_copy (Root -> R child);
Step 5 : Temp-Data = Then return.

Program code :-

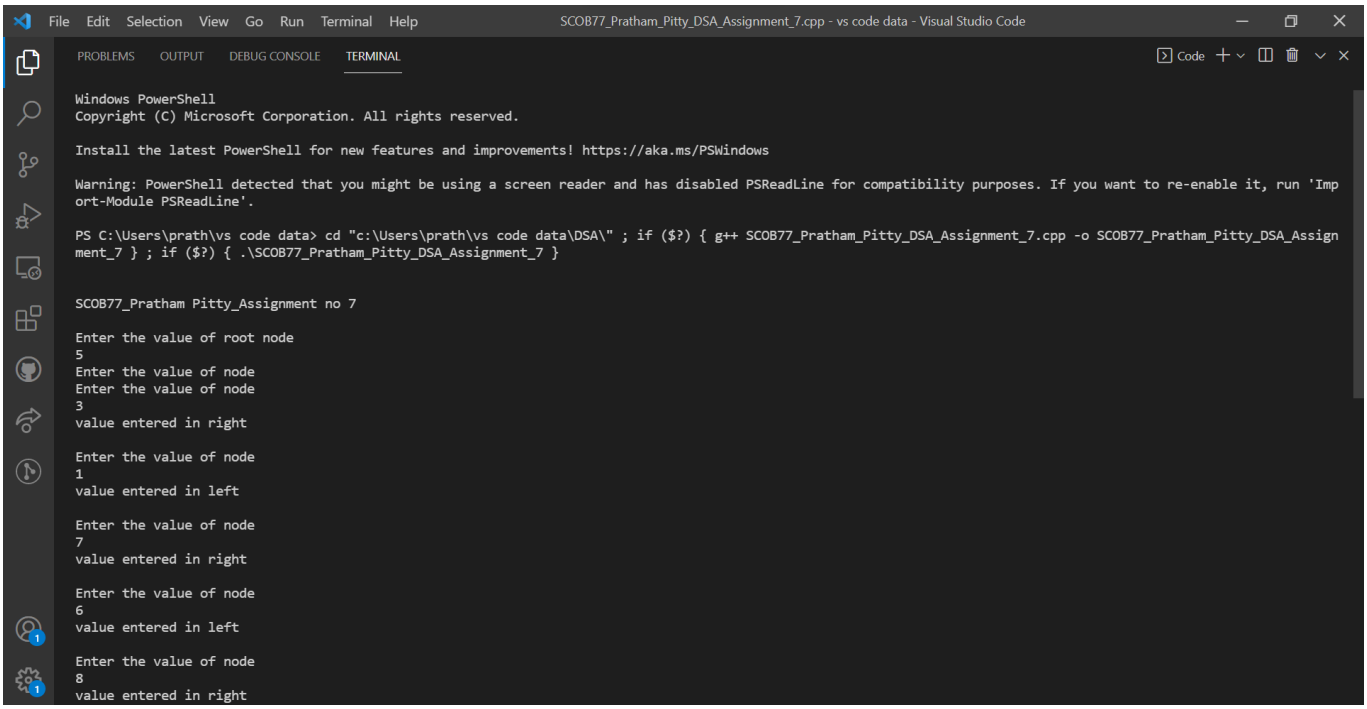
```
#include <iostream>
#include <conio.h>
using namespace std;
struct tree
{
   tree *l, *r;
   int data;
} *root = NULL, *p = NULL, *np = NULL, *q;

void create()
{
   int value, c = 0;
   while (c < 7)
   {
      if (root == NULL)
      {
         root = new tree;
         cout << "Enter the value of root node\n";
         cin >> root->data;
         root->r = NULL;
         root->l = NULL;
      }
      else
      {
         p = root;
         cout << "Enter the value of node\n";
         cin >> value;
         while (true)
```

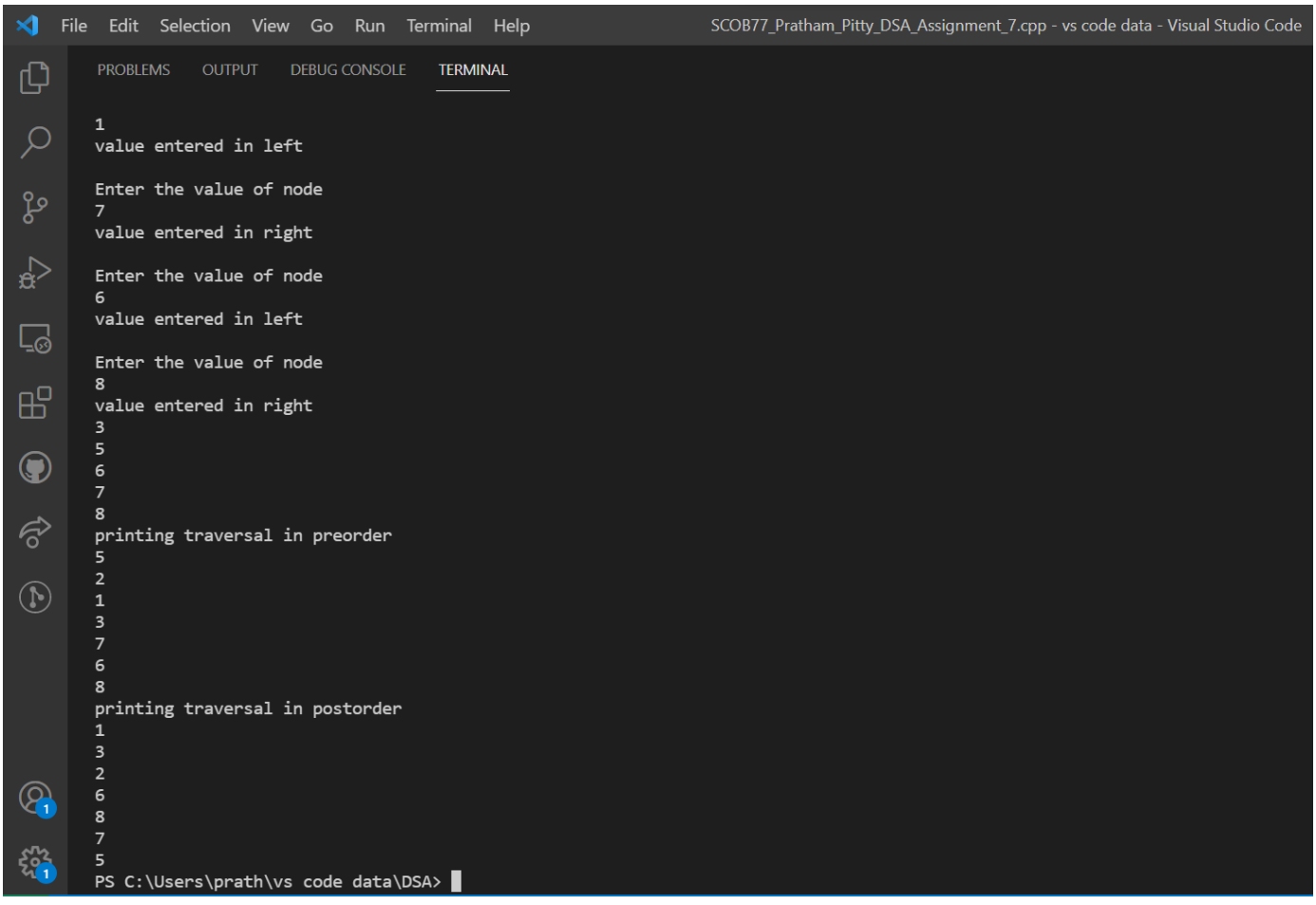
```
{
   if (value < p->data)
   {
      if (p->l == NULL)
      {
         p->l = new tree;
         p = p->l;
         p->data = value;
         p->l = NULL;
         p->r = NULL;
         cout << "value entered in left\n"<<endl;
         break;
      }
      else if (p->l != NULL)
      {
         p = p->l;
      }
   }
   else if (value > p->data)
   {
      if (p->r == NULL)
      {
         p->r = new tree;
         p = p->r;
         p->data = value;
         p->l = NULL;
         p->r = NULL;
         cout << "value entered in right\n"<<endl;
         break;
```

```
                }
                else if (p->r != NULL)
                {
                    p = p->r;
                }
            }
        }
    }
    c++;
  }
}
void inorder(tree *p)
{
  if (p != NULL)
  {
    inorder(p->l);
    cout << p->data << endl;
    inorder(p->r);
  }
}
void preorder(tree *p)
{
  if (p != NULL)
  {
    cout << p->data << endl;
    preorder(p->l);
    preorder(p->r);
  }
}
```

```
void postorder(tree *p)
{
   if (p != NULL)
   {
      postorder(p->l);
      postorder(p->r);
      cout << p->data << endl;
   }
}
int main()
{
   cout << "\n\nSCOB77_Pratham Pitty_Assignment no 7 \n\n";
   create();
   cout << "printing traversal in inorder\n";
   inorder(root);
   cout << "printing traversal in preorder\n";
   preorder(root);
   cout << "printing traversal in postorder\n";
   postorder(root);
   getch();
}
```

Output :-


File Edit Selection View Go Run Terminal Help
SCOB77_Pratham_Pitty_DSA_Assignment_7.cpp - vs code data - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows
Warning: PowerShell detected that you might be using a screen reader and has disabled PSReadLine for compatibility purposes. If you want to re-enable it, run 'Import-Module PSReadLine'.
PS C:\Users\prath\vs code data> cd "c:\Users\prath\vs code data\DSA\" ; if ($?) { g++ SCOB77_Pratham_Pitty_DSA_Assignment_7.cpp -o SCOB77_Pratham_Pitty_DSA_Assignment_7 } ; if ($?) { .\SCOB77_Pratham_Pitty_DSA_Assignment_7 }
SCOB77_Pratham Pitty_Assignment no 7
Enter the value of root node
5
Enter the value of node
Enter the value of node
3
value entered in right
Enter the value of node
1
value entered in left
Enter the value of node
7
value entered in right
Enter the value of node
6
value entered in left
Enter the value of node
8
value entered in right


File Edit Selection View Go Run Terminal Help
SCOB77_Pratham_Pitty_DSA_Assignment_7.cpp - vs code data - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1
value entered in left
Enter the value of node
7
value entered in right
Enter the value of node
6
value entered in left
Enter the value of node
8
value entered in right
3
5
6
7
8
printing traversal in preorder
5
2
1
3
7
6
8
printing traversal in postorder
1
3
2
6
8
7
5
PS C:\Users\prath\vs code data\DSA>