Institute of Computer Technology

B. Tech Computer Science and Engineering

Subject: DS (2CSE302)

PRACTICAL-15

AIM: - Implement the real-life scenario using binary tree.

Cisco Systems, Inc. is an American multinational technology, which sells networking hardware, software, telecommunications equipment, and other high-technology services and products. Roshni is working at Cisco, Ahmedabad and she wants to capture all orders of telecommunications equipment in the form of a binary tree as each order is linked to the previous one in a parent-child relationship. Here, each node is having the constraint that it has either two children or zero. Kindly perform the below operation using C language also create the binary tree structure using paper and pen:

- a. Create the binary tree of the given list of order-Id (define max-size as 7) 20, 15, 30, 25, 19, 31, 45
- b. Print root node (level-0), level-1, and level-2 node Input:

20 15 30 25 19 31 45

Output:

Root element of binary tree is: 20 Level-1 element of binary tree is: 15 30 Level-2 element of binary tree is: 25 19 31 45

Hint:

/* A binary tree node has data, pointer to left child and a pointer to right child */

```
struct node
{
    int data;
    struct node* left;
    struct node* right;
};
```

Algorithm for Implementation:

- 1. Declare a binary tree node using structure (Syntax is shown above)
- 2. Create a function newNode(int data) with return type "struct node*" to create a binary tree node.
- a. Allocate memory for the node using malloc function
- b. Assign data to the data part of the node

DATA STRUCTURE (2CSE302)

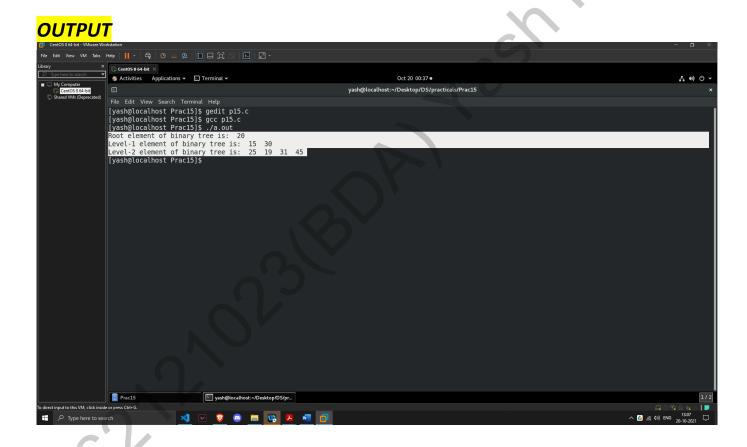
- c. Assign left and right child as NULL for every new node
- 3. Inside main(), call the function newNode(int data) and declare a "root" variable of "struct node*" type.
- 4. Then, assign a newNode(int data) to root->left and root->right and so on to create complete binary tree.
- 5. Create a function printCurrentLevel(struct node* root, int level) for levelwise element printing and call the function inside main().

SOLUTION

```
#include <stdio.h>
#include <malloc.h>
struct node {
  int yash;
  struct node *leftNode;
  struct node *rightNode;
};
struct node *newNode(int value){
  struct node *node=(struct node*)malloc(sizeof(struct node));
  node->yash=value;
  node->leftNode=NULL;
  node->rightNode=NULL;
  return node;
void printCurrentLevel(struct node *root ,int lvl){
  if(root==NULL){
    return;
  if(|v|==0)
    printf(" %d ",root->yash);
  if(|v|>0){
    printCurrentLevel(root->leftNode,lvl-1);
    printCurrentLevel(root->rightNode,lvl-1);
int main(){
  struct node *root= newNode(20);
  root->leftNode=newNode(15);
```

DATA STRUCTURE (2CSE302)

```
root->rightNode=newNode(30);
root->leftNode->leftNode=newNode(25);
root->leftNode->rightNode=newNode(19);
root->rightNode->leftNode=newNode(31);
root->rightNode->rightNode=newNode(45);
printf("Root element of binary tree is: ");
printCurrentLevel(root,0);
printf("\nLevel-1 element of binary tree is: ");
printCurrentLevel(root,1);
printf("\nLevel-2 element of binary tree is: ");
printCurrentLevel(root,2);
}
```



DATA STRUCTURE (2CSE302)