Parshvanath Charitable Trust's

A. P. SHAH INSTITUTE OF TECHNOLOGY

(Approved by AICTE New Delhi & Govt. of Maharashtra, Affiliated to University of Mumbai)
(Religious Jain Minority)

Department of Information Technology

Academic Year: 2018-19	**Name of Student: Amisha Karia**
Semester: VII	**Student ID: 15104008**
Class / Branch: BE IT	**Date of Performance:13/8/18**
Subject: Intelligent Systems	**Date of Submission:13/8/18**
Name of Instructor: Prof. Nahid Shaikh	

Experiment No. 4

Aim: To implement hill climbing algorithm.

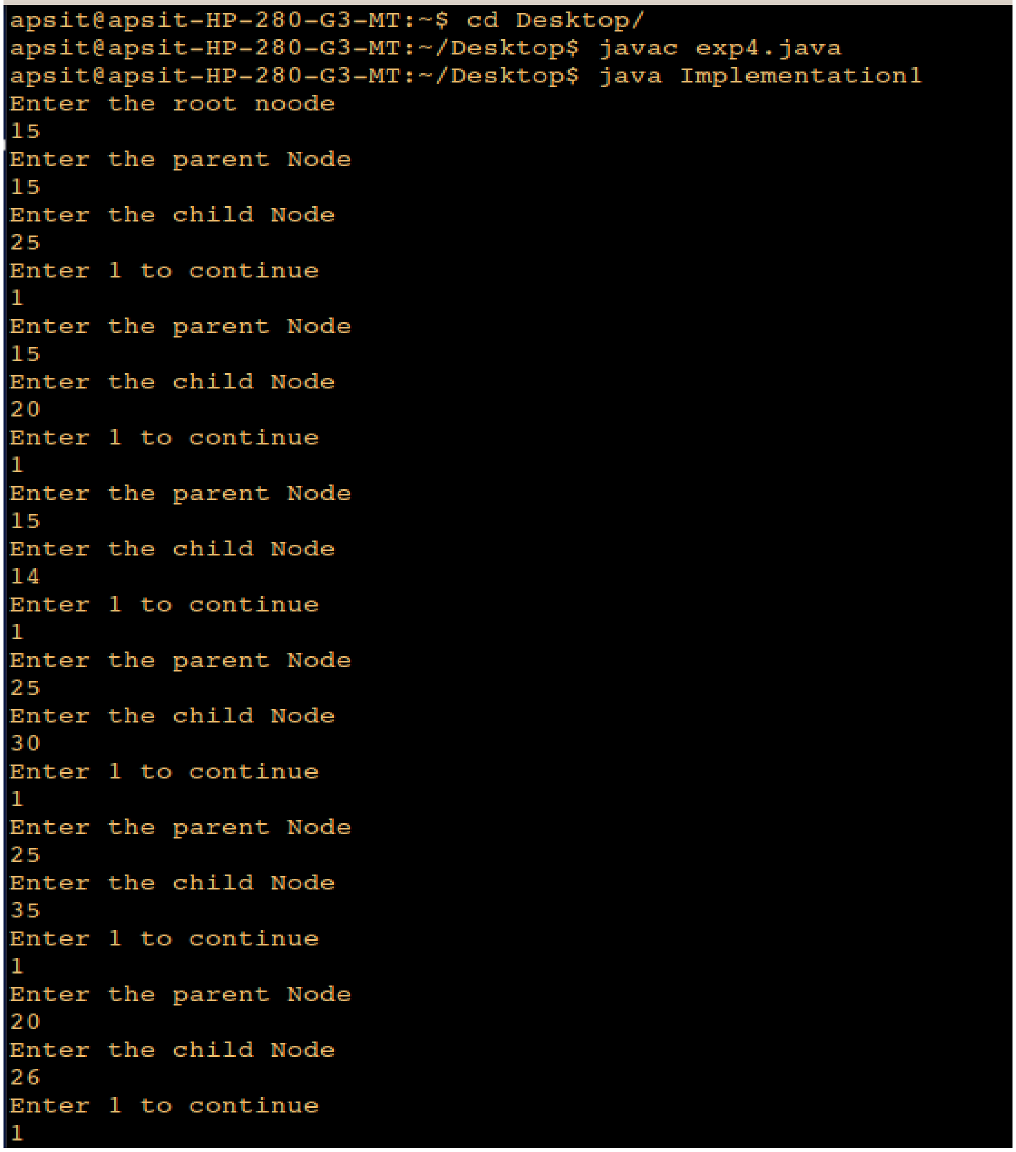
CODE:

```
import java.util.Scanner;
import java.util.ArrayList;
import java.util.LinkedList;
class TNode{
private TNode parent;
private ArrayList<TNode> children;
private int data;
public TNode(){
parent =null;
children=null;
data=-1;
}
public TNode(int data){
this.data=data;
parent=null;
children=null;
}
public TNode getParent()
{
return parent;
}
public void setParent(TNode parent)
```

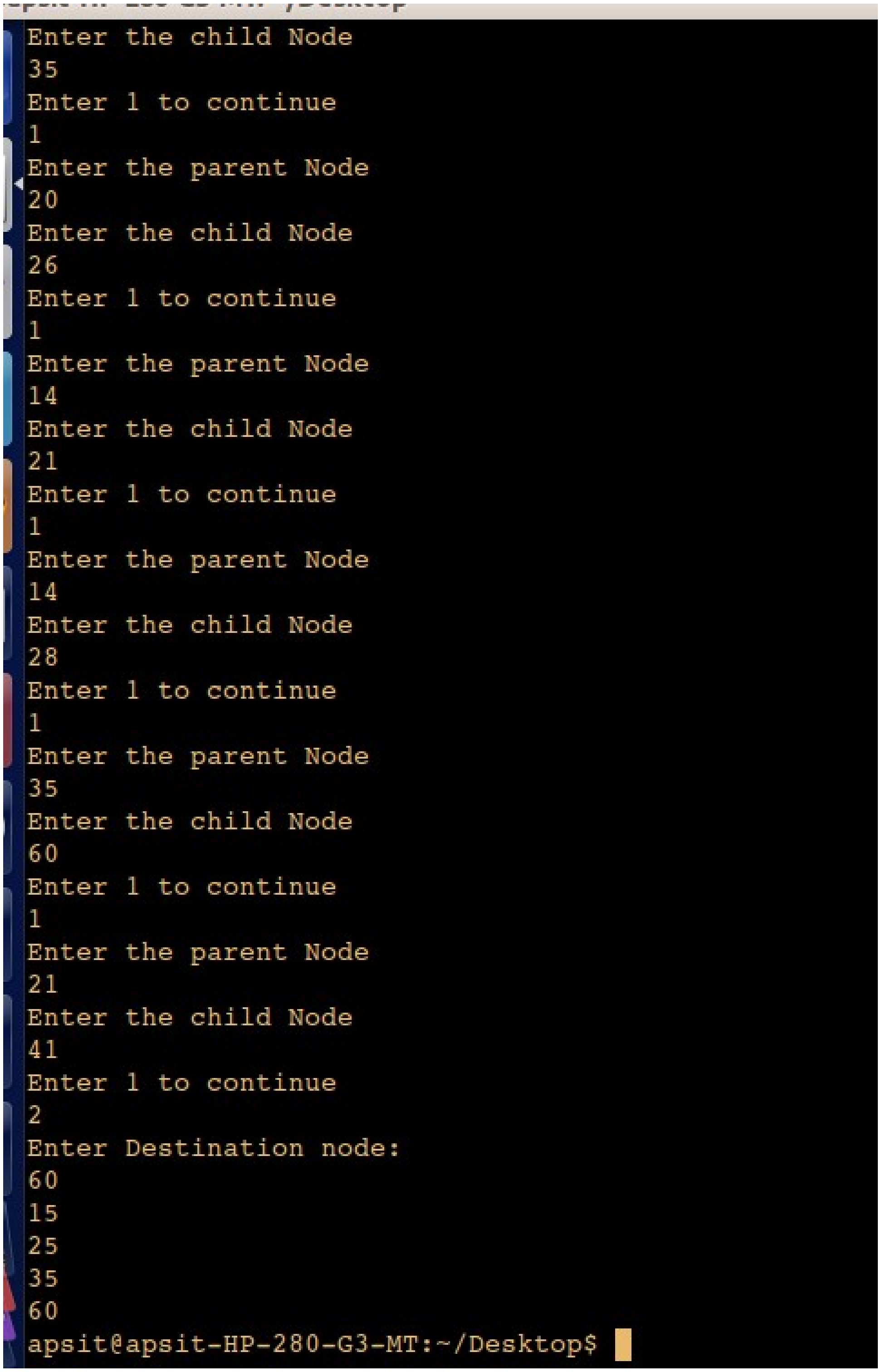
```
{this.parent=parent;
}
public ArrayList<TNode>getChildren(){
return children;
}
public void setChildren(TNode childs)
{TNode child =childs;
if(children==null){
children= new ArrayList<TNode>();
}children.add(child);
}
//Get Data
public int getData(){
return data;
}
public void setData(int data){
this.data=data;
}
//Get Number of Children
public int getNc()
{return children.size();
}
}
class Tree
{
TNode root;
public void tree(){
root=null;
}
public Tree(TNode root){
this.root=root;
}
public void addNode(TNode par, int data){
TNode ch= new TNode(data);
TNode n= root;
LinkedList<TNode> chs=new LinkedList<TNode>();
chs.add(root);
ArrayList<TNode> temp;
while(!chs.isEmpty()){
n=chs.remove();
if(n.getData()==par.getData()){
n.setChildren(ch);
ch.setParent(n);
break;
}
else
{
if((temp=n.getChildren())!=null){
for(int i=0;i<n.getNc();i++){
chs.add(temp.get(i));
}}
}
}
}
void hill(TNode root, int dest)
{
TNode myRoot=root;
while(true){
if(myRoot.getData()==dest){
System.out.println(myRoot.getData());
break;
}else if(myRoot.getData()<dest){
System.out.println(myRoot.getData());
ArrayList<TNode> children=myRoot.getChildren();
TNode max= myRoot;
if(children.size()>0)
{for(int i=0;i<children.size();i++){
if(children.get(i).getData()>max.getData()){
max=children.get(i);
}
}
myRoot=max;
}
else{
System.out.println("Not Found");
break;
}
}
else{
System.out.println("Not Found");
break;
}
}
}}
class Implementation1{
public static void main( String[] args ){
Scanner sc= new Scanner(System.in);
System.out.println("Enter the root noode");
int data =sc.nextInt(),data2;
TNode root= new TNode(data);
Tree tree=new Tree(root);
int m=1;
```

```
while(m==1)
{
System.out.println("Enter the parent Node");
data= sc.nextInt();
System.out.println("Enter the child Node");
data2=sc.nextInt();
tree.addNode(new TNode(data),data2);
System.out.println("Enter 1 to continue");
m=sc.nextInt();
}
System.out.println("Enter Destination node:");
tree.hill(root,sc.nextInt());
}
}
```

Output-



```
apsit@apsit-HP-280-G3-MT:~$ cd Desktop/
apsit@apsit-HP-280-G3-MT:~/Desktop$ javac exp4.java
apsit@apsit-HP-280-G3-MT:~/Desktop$ java Implementation1
Enter the root noode
15
Enter the parent Node
15
Enter the child Node
25
Enter 1 to continue
1
Enter the parent Node
15
Enter the child Node
20
Enter 1 to continue
1
Enter the parent Node
15
Enter the child Node
14
Enter 1 to continue
1
Enter the parent Node
25
Enter the child Node
30
Enter 1 to continue
1
Enter the parent Node
25
Enter the child Node
35
Enter 1 to continue
1
Enter the parent Node
20
Enter the child Node
26
Enter 1 to continue
1
```



```
Enter the child Node
35
Enter 1 to continue
1
Enter the parent Node
20
Enter the child Node
26
Enter 1 to continue
1
Enter the parent Node
14
Enter the child Node
21
Enter 1 to continue
1
Enter the parent Node
14
Enter the child Node
28
Enter 1 to continue
1
Enter the parent Node
35
Enter the child Node
60
Enter 1 to continue
1
Enter the parent Node
21
Enter the child Node
41
Enter 1 to continue
2
Enter Destination node:
60
15
25
35
60
apsit@apsit-HP-280-G3-MT:~/Desktop$ 
```

Conclusion:Hence we have implemented a program to get the knowledge of hill climbing algorithm.We also studied the concept of local maximum,global maximum, flat local maximum,etc.