**Name # Saad Iqbal**

**REG # 32903**

**Class #BEE-6B**

**Lab no# 10**

#include<iostream>//libraries

#include<vector>

#include<algorithm>

#include<math.h>

using namespace std;//defining std

class heaptree{//class of heaptree

vector<int> htree;//private member variable

public://public functions

heaptree(){//constructor

htree.push\_back(-1);

}

~heaptree(){}//destructor

int size(){//return size of heap tree

return htree.size()-1;

}

bool isEmpty(){//check tree,if empty or not

if (htree.size() == 1)

return true;

else

return false;

}

void insert(int b){//insert value in tree

htree.push\_back(b);

for (int v = htree.size() - 1; v > 1; v=v / 2)

{

if (htree[v] < htree[v / 2])

swap(htree[v], htree[v / 2]);

else

v = 1;

}

}

void POP(){//removing min value from tree

htree[1] = htree[size()];

htree.pop\_back();

for (int h = 1; 2 \* h <= size();){

if ((2 \* h) + 1 <= size()){

if (htree[2 \* h] < htree[(2 \* h) + 1]){

if (htree[2 \* h] < htree[h]){

swap(htree[h], htree[2 \* h]);

h \*= 2;

}

else

h = size() + 1;

}

else{

if (htree[(2 \* h) + 1] < htree[h]){

swap(htree[h], htree[(2 \* h) + 1]);

h = (h \* 2) + 1;

}

else

h = size() + 1;;

}

}

else{

if (htree[h]>htree[2 \* h]){

swap(htree[h], htree[2 \* h]);

h \*= 2;

}

else{

h = size() + 1;

}

}

}

}

vector<int> min\_heap\_tree(vector<int> &a){//turn random array into heap tree

int x = a.size();

htree.clear();

htree.push\_back(-1);

while (x){

insert(a[x - 1]);

x--;

}

return htree;

}

int top(){//return root value/node

if (!isEmpty())

return htree[1];

else

return -1;

}

int height(){//return height of tree

if (isEmpty())

return -1;

else

return log2(size());

}

void print(){//print tree

for (int i = 0; i <= size(); i++){

cout << htree[i] << " ";

}

cout << endl;

}

void heap\_sort(vector<int> a){//sort the random array using heap tree

this->min\_heap\_tree(a);

vector<int> c;

while (!this->isEmpty()){

c.push\_back(this->top());

this->POP();

}

htree.clear();

htree = c;

}

};

int main(){

heaptree x;

vector<int> array1 = {4,2,6,3,88,55,32,64,24,43,1,75,23,13,42,12,56,53,67,5,65,7,85,97,8,95,45,87};//random array

x.min\_heap\_tree(array1);

cout << "In heap tree"<<endl;

x.print();

cout << "Size of heap" << endl;

cout<<x.size()<<endl;

cout << "Height of tree" << endl;

cout << x.height() << endl;

cout << "Root or min value in tree" << endl;

cout << x.top() << endl;

cout << "Check empty case" << endl;

cout << x.isEmpty() << endl;

x.POP();

cout << "After a single pop" << endl;

x.print();

x.heap\_sort(array1);

cout << "After heap sort" << endl;

x.print();

system("pause");

return 0;

}