**Heap Help:**

#include <iostream>

#include <math.h>

using namespace std;

void swap(int &a, int &b)

{

int temp = a;

a = b;

b = temp;

}

class Heap

{

public:

int values[30];

int size;

public:

Heap()

{

size = 0;

}

bool isEmpty()

{

return size == 0;

}

bool isFull()

{

return size == 30;

}

int top()

{

if (isEmpty())

{

cout << "Heap is empty!\n";

return -1;

}

else

{

//write your code here

}

}

void push(int value)

{

//write your code here

}

void heapify(int index)

{

//write your code here

}

void buildheap()

{

for (int i = size / 2; i >= 1; i--)

{

heapify(i);

}

}

void buildheapfromarray(int\* arr, int size)

{

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

push(arr[i]);

buildheap();

}

void deleteMin()

{

//write your code here

buildheap();

}

int pop(int index)

{

if (index < 1 || index > size)

{

cout << "Invalid index!\n";

return -1;

}

else

{

//write your code here

}

}

void print()

{

if (isEmpty())

{

cout << "Heap is empty!\n";

return;

}

for (int i = 1; i <= size; i++)

{

cout << values[i] << " ";

if (i==1 || i==3 || i==7 || i==15)

cout << endl;

}

cout << endl;

}

int getSize()

{

//write your code here

}

int getHeight()

{

//write your code here

}

};

int main()

{

Heap myHeap = Heap();

const int arraySize = 15;

int array[arraySize];

for (int i = 0; i < arraySize; i++)

array[i] = (rand() % 100) + 1;

myHeap.buildheapfromarray(array, arraySize);

myHeap.print();

// Create a Menu for all functions

getchar();

return 0;

}