Program to Sort an Array of Integers in Ascending Order Using Heap Sort

Student Name: Neha Sharma
UID:20BCS4576
Branch: CSE-IOT
Section/Group: 20BIT-1/A
Semester: 3rd
Date of Performance: 20/11/2021
Subject Name: Data Structures Lab
Subject Code: 20CSP-236

1. Aim/Overview of the practical: Program to Sort an Array of Integers in Ascending Order Using Heap Sort

2. Task to be done: To write a program for sorting the array of an elements in ascending order.

3. Algorithm:

Step 1: Start

Step 2: Build a max heap from the input data.

Step 3: At this point, the largest item is stored at the root of the heap. Replace it with the last item of the heap followed by reducing the size of heap by 1. Finally, heapify the root of the tree.

Step 4: Repeat step 2 while the size of the heap is greater than 1.

Step 5: End

4. Steps for experiment/practical:

```
#include <iostream>
using namespace std;
void heapify(int arr[], int n, int i)
{
int largest = i; // Initialize largest as root
int l = 2 * i + 1; // left = 2*i + 1
int r = 2 * i + 2; // right = 2*i + 2
if (l < n && arr[l] > arr[largest])
largest = l;
if (r < n && arr[r] > arr[largest])
largest = r;
if (largest != i) {
swap(arr[i], arr[largest]);
heapify(arr, n, largest);
}
}
void heapSort(int arr[], int n)
{
for (int i = n / 2 - 1; i >= 0; i--)
heapify(arr, n, i);
for (int i = n - 1; i > 0; i--) {
swap(arr[0], arr[i]);
heapify(arr, i, 0);
}
}
void printArray(int arr[], int n)
{
for (int i = 0; i < n; ++i)
cout << arr[i] << " ";
cout << "\n";
```

```
}
int main()
{
int arr[] = { 12, 11, 13, 5, 6, 7 };
int n = sizeof(arr) / sizeof(arr[0]);
heapSort(arr, n);
cout <<"Neha Sharma";
cout << "\nSorted array is \n";
printArray(arr, n);
}
```

5. Output:


input
Neha Sharma
Sorted array is
5 6 7 11 12 13
...Program finished with exit code 0
Press ENTER to exit console.

Learning outcomes (What I have learnt):

1. Program to Sort an Array of Integers in Ascending Order Using Heap Sort
2. Respective syntax and implementations of functions and classes used for heap sort.

Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			