Assignment 6

Aim :

Read the marks obtained by students of second year in an online examination of particular subject. Find out maximum and minimum marks obtained in that subject using heap data structure.

Objective :

To find maximum and minimum marks obtained by the students in second year in a particular subject using a binary heap (either max heap or min heap) and then sorting the heap using heap sort algorithm for desired output.

Theory :

A binary heap is a complete binary tree which satisfies the heap ordering property.

The ordering can be one of two types:

- the min-heap property: the value of each node is greater than or equal to the value of its parent, with the minimum-value element at the root.
- the max-heap property: the value of each node is less than or equal to the value of its parent, with the maximum-value element at the root.

  We create a heap by adding numbers from left to right level by level. Heap can be implemented using an array or a priority queue.

  For sorting the heap, after it's creation, the first position of the array would contain either the smallest or the largest element depending on whether max heap or min heap is created ,heap sort algorithm swaps the first element in the heap with the last one and heapify the heap excluding the last element and reduce the size of the array by one. Repeat the steps until the complete heap is sorted.

Code :

```
#include <iostream>
using namespace std;
// To heapify a subtree rooted with node i which is
// an index in arr[]. n is size of heap
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 left child is larger than root
	if (l < n && arr[l] > arr[largest])
		largest = l;
	// If right child is larger than largest so far
	if (r < n && arr[r] > arr[largest])
		largest = r;
	// If largest is not root
	if (largest != i)
	{
		swap(arr[i], arr[largest]);
		// Recursively heapify the affected sub-tree
		heapify(arr, n, largest);
	}
}
// main function to do heap sort
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);
        }
}
int main()
{
   int n,arr[100];
        cout<<"Enter the no. of student's marks you want to enter. :\n";
        cin>>n;
        cout<<"Enter the marks :\n";
        for(int i=0;i<n;i++)
   {
      cin>>arr[i];
   }
        heapSort(arr, n);
        cout << "The maximum marks are: "<<arr[n-1]<<"\n";
        cout<<"The minimum marks are: "<<arr[0]<<"\n";
}
```

Output:

```
Enter the no. of student's marks you want to enter. :
5
Enter the marks :
```

```
98
68
88
77
89
The maximum marks are: 98
The minimum marks are: 68
```

Conclusion-:

The objective of creating a heap from an array was completed. The heap was then sorted to give out the maximum and minimum marks obtained by students using heap sort algorithm.