**Program – 9**

**Aim – Write an algorithm and program to implement heap sort.**

**Algorithm:**

**Step 1**: [Build Heap H]  
Repeat for i=0 to N-1  
CALL INSERT\_HEAP(ARR, N, ARR[i])  
[END OF LOOP]

**Step 2**: Repeatedly Delete the root element  
Repeat while N > 0  
CALL Delete\_Heap(ARR,N,VAL)  
SET N = N+1  
[END OF LOOP]

**Step 3**: END

**Source Code:**

#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 << "Sorted array is \n";

printArray(arr, n);

}