```java
package heap;

/*Algorithm:
    1. Arrange the array elements into a heap following BFS.
    2. Form max-heap -- heapify if required.
    3. Exchange the top root(i.e, arr[0] having max value of all) with the last leaf node (minimum of all).
    4. Remove the last node (which has max value now).
    5. Repeat from step 2.
*/
public class HeapSort {

    void sort(int arr[],int n) {
        int lastParent = n/2 - 1;  //last parent node which has leaf node.

        for(int i = lastParent; i>=0; --i) {
            heapify(arr,n,i);   // careful in passing the last value  n and i
        }

        // the nodes will be removed from the heap once it delivers the max value to the leaf node.
        int lastElement = n-1;      // This for loop is to fetch the max from a[0] and exchange it with the last leaf node having min value. Then arrange the elements using heapify.
        for(int i = lastElement; i>0; --i) {
            int temp = arr[0];  // replacing the max(root element) with the last leaf(min value) node
            arr[0] = arr[i];
            arr[i] = temp;

            heapify(arr,i,0);  //rearrange the elements inside remaining heap elements
            // careful in passing last 2 values i and 0
        }
    }

    static void heapify(int[] arr,int n, int i) {  // to arrange the elements into the tree to maintain max heap
        int largest = i;
        int left = 2*i + 1;
        int right = 2*i + 2;

        if(left < n && arr[left] > arr[largest])
            largest = left;

        if(right < n && arr[right] > arr[largest])
            largest = right;

        if(largest != i) {
            int temp = arr[i];
            arr[i] = arr[largest];
            arr[largest] = temp;

            heapify(arr,n,largest);  // for arranging the affected nodes;
        }
    }
    static void printHeap(int[] arr, int n) {
        System.out.println("Sorted elements are:");
        for(int i = 0;i < n ; ++i)
```

```
53              System.out.print(arr[i] + " ");
54      }
55      public static void main(String[] args) {
56
57          HeapSort hs = new HeapSort();
58          int arr[] = {12,11,13,5,6,7 };
59          int n = arr.length;
60          hs.sort(arr, n);
61          printHeap(arr,n);
62
63      }
64
65  }
66
67  /*
68   ----------------------- Output ----------------------
69
70  Sorted elements are:
71  5 6 7 11 12 13
72
73  */
74
```