Assignment 2: Heapsort

Implement the heapsort algorithm described in the lecture.

Direction:

Create a class named `Heapsort` that contains a method named `sort`. This method will take an integer array as parameter. All the changes should be made in the array passed as parameter. Nothing needs to be returned. You can add any number of other methods you feel necessary in that class. The input size of the array will be no more than 100000.

Way to Solution

You know about the heapify operation, where an unordered array is turned into a heap-ordered array. First implement that. This can be done using only the input array. Then you can turn the updated array to a sorted one. Again you will need the array only. No extra memory should be used. Go through the reading materials to find out the algorithm for that (Listed at the end of the document).

Sample Input/Output:

Some set of input and output are given here. Format is explained with the example below.

```
10
10 9 8 4 6 2 7 1 3 5
```

The first line contains how many numbers are there, followed by a line that contains all the numbers separated by space(s).

Also for the above case the numbers are following max-heap ordering. Some more are given in the table below.

Set	Sample Input	Heap

		Ordered?
1	50 44 11 40 13 45 12 47 17 23 43 21 37 2 6 41 25 33 28 16 38 26 48 15 19 27 14 29 35 5 50 9 3 46 22 10 31 42 7 20 36 18 24 49 8 4 34 1 32 39 30	No
2	50 50 49 44 47 48 40 32 45 43 41 46 34 35 18 30 31 42 39 33 36 37 38 28 19 10 22 1 5 15 13 20 17 2 7 21 14 25 9 23 27 24 3 29 6 26 12 4 16 11 8	Yes
3	20 97 89 87 84 78 76 71 76 60 60 76 51 42 33 65 60 25 19 12 4	Yes
4	20 89 23 99 1 61 12 92 81 71 93 19 84 42 31 9 57 50 35 76 70	No

Sample Output for Set 1 and 2:

```
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29
30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50
```

Sample Output for Set 3:

```
4 12 19 25 33 42 51 60 60 60 65 71 76 76 76 78 84 87 89 97
```

Sample Output for set 4:

```
1 9 12 19 23 31 35 42 50 57 61 70 71 76 81 84 89 92 93 99
```

Evaluation

Submission will be evaluated using following criteria (with distribution of marks)

Criteria	Marks
Created necessary java files (according to directions)	5
Sorting a heap-ordered input (correct)	40
Establishing heap ordering (correct)	40
Sorting (in general)	15
Total	**100**

Sorting a heap-ordered input (incorrect)*****	5-30
Establishing heap ordering (incorrect)*****	5-30

***** Marking depends on evaluator.

Notes on copying other's code: If anybody found copying solution code from other student(s), all of them will be penalized. Penalty includes

- ***Assigning ZERO as mark for the solution submitted***
- ***Assigning ZERO as mark for the best submission among all other submissions. (assigned at the end of semester)***

Copying solutions from the Internet will also incur similar penalties.

Reading Materials:

- *Computer Algorithms*, By Ellis Horowitz , Sartaj Sahni, Sanguthevar Rajasekaran. ISBN: 9780929306414 **Page-92**
- *Introduction to Algorithms*, By Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein. ISBN: 9780262033848 **Page-127**