Heap and Priority Queue:

1. You are required to write two programs, each of which has the same function, but a different underlying algorithm. The programs take as input a file, containing a list of household salaries. The first entry in the file indicates the number of families, followed by the household income of each of these families (each of these entries is separated by a newline). The program reads this file and asks for an input integer k, less than the total number of salaries. The program then outputs the k lowest salaries in ascending order. The three programs implement this functionality in the following ways:
    - The first program makes a heap of the numbers by inserting the numbers one at a time into the heap (William's method) and removing the k minimum numbers from the heap (while maintaining the heap).
    - The second program builds a heap bottom up (Floyd's method) and removes the k minimum numbers from the heap (while maintaining the heap).

    Implement these two algorithms in two different programs. To Test

    - Create a big (say one million salaries) Test case with random number generator in where salaries are in certain range and unique. Compare the runtime of both the program. Print the execution time taken by your program in both case.

2. Harshit has recently learned "Binary Heap" and implemented the following operations,
   (i) insert element to heap
   (ii) get the minimum value from the heap
   (iii) extract the minimum element from the heap.

   To learn this data structure better, Harshit took an empty heap and applied some operations. Also, he wrote down all the operations for logging purpose, in the following format
   (i) insert x = inserted the element x into heap.
   (ii) getmin x = the value of minimum element in heap was x.
   (iii) removemin = deleted the minimum element from the heap. (delete only one instance, if many minimum elements) [10]

   Constraints:
   1) At any time, the heap can contain many integers or none, where some might be equal.
   2) When getmin / removemin is applied, the heap is guaranteed to have at least one element.

   When harshit was away for dinner, his little sister(Khyati) came and erased few operations in the log.

   Now, Khyati is worried about making Harshit's sequence inconsistent. For example, if we apply operations in the order of log, results of getMin operations might differ from the results of Harshit, and some of getMin or removeMin operations may be inconsistent, as the heap is empty when applied.

Now, Khyati wants to add some operations in log to make them consistent. (result of getmin should match with the result in the record and heap is non-empty during getmin and removemin)
Now, Khyati wants to fix this asap and she is asking you to add least possible number of operations to the log.

Note: arbitrary number of operations can be added at the beginning of log/between operations / end of the log.

Input Format:
first line will have n(1<= n <= 1,00,000), current number of records in Harshit's log
The following "n" line are the operations described in the above format .
All the numbers in input, will be integers not exceeding 10^9.

Output Format:
first line print m, minimum possible number of records in modified sequence.
Next m lines, describe the corrected sequence of operations in the above format.(one per line)

Note: the input sequence must be the subsequence of the output sequence.

It is guaranteed that, there exists one correct answer.

Sample Input and Output:
Test 1:

Input
2
insert 6
getMin 7

Output
4
insert 6
removeMin
insert 7
getMin 7

Test 2:
Input
4
insert 6
insert 6
removeMin
getMin 9

Output

6
insert 6
insert 6
removeMin
removeMin
insert 9
getMin 9

3. You are given an array on n integers. You need to find the kth smallest number from the array without sorting the elements and using a heap of size k.
4. Write a function that merges two max-heaps into one max-heap.