

Задача А. Пирамида ли? (1 балл)

Имя входного файла: `isheap.in`
Имя выходного файла: `isheap.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Структуру данных неубывающая пирамида можно реализовать на основе массива.

Для этого должно выполняться *основное свойство неубывающей пирамиды*, которое заключается в том, что для каждого $1 \leq i \leq n$ выполняются условия:

- если $2i \leq n$, то $a[i] \leq a[2i]$;
- если $2i + 1 \leq n$, то $a[i] \leq a[2i + 1]$.

Дан массив целых чисел. Определите, является ли он неубывающей пирамидой.

Формат входного файла

Первая строка входного файла содержит целое число n ($1 \leq n \leq 10^5$). Вторая строка содержит n целых чисел по модулю не превосходящих $2 \cdot 10^9$.

Формат выходного файла

Выведите «YES», если массив является неубывающей пирамидой, и «NO» в противном случае.

Пример

<code>isheap.in</code>	<code>isheap.out</code>
5 1 0 1 2 0	NO
5 1 3 2 5 4	YES

Задача В. Пирамидальная сортировка (2 балла)

Имя входного файла: `sort.in`
Имя выходного файла: `sort.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дан массив целых чисел. Ваша задача — отсортировать его в порядке неубывания с помощью пирамидальной сортировки (heap sort). За решения, основанные на любых других сортировках, баллы ставиться не будут.

Формат входного файла

В первой строке входного файла содержится число n ($1 \leq n \leq 100000$) — количество элементов в массиве. Во второй строке находятся n целых чисел, по модулю не превосходящих 10^9 .

Формат выходного файла

В выходной файл надо вывести этот же массив в порядке неубывания, между любыми двумя числами должен стоять ровно один пробел.

Пример

<code>sort.in</code>	<code>sort.out</code>
10 1 8 2 1 4 7 3 2 3 6	1 1 2 2 3 3 4 6 7 8

Примечание

Необходимо написать свою сортировку кучей.

Задача С. Цифровая сортировка (2 балла)

Имя входного файла: `radixsort.in`
Имя выходного файла: `radixsort.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дано n строк, выведите их порядок после k фаз цифровой сортировки.

Формат входного файла

В первой строке входного файла содержится число n — количество строк, m — их длина и k — число фаз цифровой сортировки ($1 \leq n \leq 1000$, $1 \leq k \leq m \leq 1000$). В следующих n строках находятся сами строки.

Формат выходного файла

Выведите строки в порядке в котором они будут после k фаз цифровой сортировки.

Пример

<code>radixsort.in</code>	<code>radixsort.out</code>
3 3 1 bbb aba baa	aba baa bbb
3 3 2 bbb aba baa	baa aba bbb
3 3 3 bbb aba baa	aba baa bbb

Примечание

Необходимо написать свою цифровую сортировку.

Задача D. Приоритетная очередь (3 балла)

Имя входного файла: `priorityqueue.in`
Имя выходного файла: `priorityqueue.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Реализуйте приоритетную очередь. Ваша очередь должна поддерживать следующие операции: добавить элемент, извлечь минимальный элемент, уменьшить элемент, добавленный во время одной из операций.

Все операции нумеруются по порядку, начиная с единицы. Гарантируется, что размер очереди в процессе выполнения команд не превысит 10^6 элементов.

Формат входного файла

Входной файл содержит описание операций с очередью. Операции могут быть следующими:

- `push x` — требуется добавить элемент x в очередь.
- `extract-min` — требуется удалить из очереди минимальный элемент и вывести его в выходной файл. Если очередь пуста, в выходной файл требуется вывести звездочку `*`.
- `decrease-key x y` — требуется заменить значение элемента, добавленного в очередь операцией `push` в строке входного файла номер x , на y . Гарантируется, что на строке x действительно находится операция `push`, что этот элемент не был ранее удален операцией `extract-min`, и что y меньше, чем предыдущее значение этого элемента.

В очередь помещаются и извлекаются только целые числа, не превышающие по модулю 10^9 .

Формат выходного файла

Выведите последовательно результат выполнения всех операций `extract-min`, по одному в каждой строке выходного файла. Если перед очередной операцией `extract-min` очередь пуста, выведите вместо числа звездочку `*`.

Пример

<code>priorityqueue.in</code>	<code>priorityqueue.out</code>
<code>push 3</code>	<code>2</code>
<code>push 4</code>	<code>1</code>
<code>push 2</code>	<code>3</code>
<code>extract-min</code>	<code>*</code>
<code>decrease-key 2 1</code>	
<code>extract-min</code>	
<code>extract-min</code>	
<code>extract-min</code>	

Примечание

Необходимо написать свою структуру данных.