

Задача А. Хип ли?

Имя входного файла: `isheap.in`
Имя выходного файла: `isheap.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Структуру данных Heap можно реализовать на основе массива.

Для этого должно выполняться *основное свойство Heap'a*, которое заключается в следующем. Для каждого $1 \leq i \leq n$ выполняются следующие 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 Heap'ом.

Формат входных данных

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 выходных данных

Выведите «YES», если массив является Heap'ом и «NO» в противном случае.

Примеры

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

Задача В. Хипуй!

Имя входного файла: `heap.in`
Имя выходного файла: `heap.out`
Ограничение по времени: 3 секунды
Ограничение по памяти: 64 мегабайта

В этой задаче вам необходимо организовать структуру данных `Heap` для хранения целых чисел, над которой определены следующие операции:

- `Insert(X)` — добавить в `Heap` число X ;
- `Extract` — достать из `Heap` наибольшее число (удалив его при этом).

Формат входных данных

Во входном файле записано количество команд N ($1 \leq N \leq 100\,000$), потом последовательность из N команд, каждая в своей строке.

Каждая команда имеет такой формат: „0 <число>“ или „1“, что означает соответственно операции `Insert(<число>)` и `Extract`. Добавляемые числа находятся в интервале от 1 до 10^7 включительно.

Гарантируется, что при выполнении команды `Extract` в структуре находится по крайней мере один элемент.

Формат выходных данных

В выходной файл для каждой команды извлечения необходимо вывести число, полученное при выполнении команды `Extract`.

Примеры

<code>heap.in</code>	<code>heap.out</code>
7	100
0 100	50
0 10	
1	
0 5	
0 30	
0 50	
1	

Задача С. Сортировка

Имя входного файла: `sort.in`
Имя выходного файла: `sort.out`
Ограничение по времени: 2.5 секунды
Ограничение по памяти: 128 мегабайт

Дан массив целых чисел. Ваша задача — отсортировать его в порядке неубывания.

Формат входных данных

В первой строке входного файла содержится число N ($1 \leq N \leq 100\,000$) — количество элементов в массиве. Во второй строке находятся N целых чисел, по модулю не превосходящих 10^9 .

Формат выходных данных

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

Примеры

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

Замечание

При решении задачи нельзя использовать встроенные функции `sort` и `sorted`.

Задача D. k -я порядковая статистика

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

Дан массив, содержащий n целых чисел. Вам нужно найти в этом массиве k -й по счету минимальный элемент ($k = 0..n - 1$), то есть элемент, который после сортировки массива по неубыванию окажется на k -м месте от начала массива (индексация элементов начинается с нуля). Решение должно иметь сложность $O(n)$ в среднем.

Элементы массива a_i задаются при помощи псевдослучайного генератора по формуле: $a_i = (1103515245a_{i-1} + 12345) \bmod 2^{31}$, то есть все элементы массива задаются одним начальным значением a_0 . Для заполнения элементов массива начальными значениями следует использовать следующую функцию:

```
void fill(vector<int> & a, int n, int start)
{
    a.resize(n);
    a[0] = start;
    for (int i = 1; i < n; ++i)
        a[i] = (1103515245LL * a[i - 1] + 12345) % (1LL << 31);
}
```

Формат входных данных

Программа получает на вход три целых числа n , a_0 и k .

n — количество элементов в массиве, $1 \leq n \leq 2 \cdot 10^7$.

a_0 — значение первого элемента в массиве, $0 \leq a_0 < 2^{31}$.

k — индекс искомого элемента, $0 \leq k < n$.

Формат выходных данных

Программа должна вывести одно целое число — k -й минимум в данной последовательности.

Примеры

<code>kth.in</code>	<code>kth.out</code>
5 123456789 2	850994577

Замечание

В примере из условия массив (до сортировки) имеет вид $\{123456789, 231794730, 1126946331, 1757975480, 850994577\}$.

Задача Е. Минимум в окне

Имя входного файла: `window.in`
Имя выходного файла: `window.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Дан ряд из N чисел. Требуется вывести минимумы из каждых K последовательных чисел.

Формат входных данных

В первой строке вводится одно натуральное число N , не превосходящее 100000. Во второй строке вводится одно натуральное число K , не превосходящее N . В следующих N строках вводится по одному натуральному числу, не превосходящему 100000.

Формат выходных данных

Выведите $N - K + 1$ чисел: минимум из первых K чисел, минимум из следующих K чисел (начиная со второго) и т.д.

Примеры

<code>window.in</code>	<code>window.out</code>
5	3
3	2
5	1
3	
5	
2	
1	