

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 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 Heap'ом 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. Хипуй!

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

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 входного файла

Во входном файле записано количество команд N ($1 \leq N \leq 100000$), потом последовательность из N команд, каждая в своей строке. Каждая команда имеет такой формат: "0 <число>" или "1", обозначающие соответственно операции **Insert**(<число>) и **Extract**. Гарантируется, что при выполнении команды **Extract** в структуре находится по крайней мере один элемент.

Формат выходного файла

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

Пример

<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. Мячик

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

На вершине лесенки, содержащей N ступенек, находится мячик, который начинает прыгать по ним вниз, к основанию. Мячик может прыгнуть на следующую ступеньку, на ступеньку через одну или через 2. (То есть, если мячик лежит на 8-ой ступеньке, то он может переместиться на 5-ую, 6-ую или 7-ую.) Определить число всевозможных маршрутов мячика с вершины на землю.

Формат входного файла

Во входном файле находится одно число $1 \leq N \leq 30$.

Формат выходного файла

В выходной файл выведите одно число — количество маршрутов.

Пример

ball.in	ball.out
4	7

Задача D. Биномиальные коэффициенты

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

В прямоугольной таблице $N \times M$ в начале игрок находится в левой верхней клетке. За один ход ему разрешается перемещаться в соседнюю клетку либо вправо, либо вниз (влево и вверх перемещаться запрещено). Посчитайте, сколько есть способов у игрока попасть в правую нижнюю клетку.

Формат входного файла

Во входном файле находится два числа NM ($1 \leq N, M \leq 10$).

Формат выходного файла

В выходной файл выведите одно число — количество маршрутов.

Пример

coeff.in	coeff.out
2 3	3
3 3	6

Задача E. Тестирующая система

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

Юный программист Саша написал свою первую тестирующую систему. Он так обрадовался тому, что она скомпилировалась, что решил пригласить школьных друзей на свой собственный контекст.

Но в конце тура выяснилось, что система не умеет сортировать команды в таблице результатов. Помогите Саше реализовать эту сортировку.

Команды упорядочиваются по правилам АСМ:

- по количеству решённых задач в порядке убывания;

- при равенстве количества решённых задач — по штрафному времени в порядке возрастания;
- при прочих равных — по номеру команды в порядке возрастания.

Формат входного файла

Первая строка содержит натуральное число n ($1 \leq n \leq 100\,000$) — количество команд, участвующих в контексте. В i -й из следующих n строк записано количество решённых задач S ($1 \leq S \leq 100$) и штрафное время T ($1 \leq T \leq 100\,000$) команды с номером i .

Формат выходного файла

В выходной файл выведите n чисел — номера команд в отсортированном порядке.

Пример

ejudge.in	ejudge.out
5	5 2 1 3 4
3 50	
5 720	
1 7	
0 0	
8 500	