

## 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 пирамида (heap) можно реализовать на основе массива.

Для этого должно выполняться *основное свойство пирамиды*, 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 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 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

## Задача С. Двоичный поиск (!) (1 балл)

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

Дан массив из  $n$  элементов, упорядоченный в порядке неубывания, и  $m$  запросов: найти первое и последнее вхождение числа в массив.

### Формат входного файла

В первую строке входного файла содержится одно число  $n$  — размер массива. ( $1 \leq n \leq 100000$ ). Во второй строке находится  $n$  чисел в порядке неубывания — элементы массива. В третьей строке находится число  $m$  — количество запросов. В следующей строке находится  $m$  чисел — запросы.

### Формат выходного файла

Для каждого запроса выведите в отдельной строке номер первого и последнего вхождения этого числа в массив. Если числа в массиве нет, выведите два раза -1.

### Пример

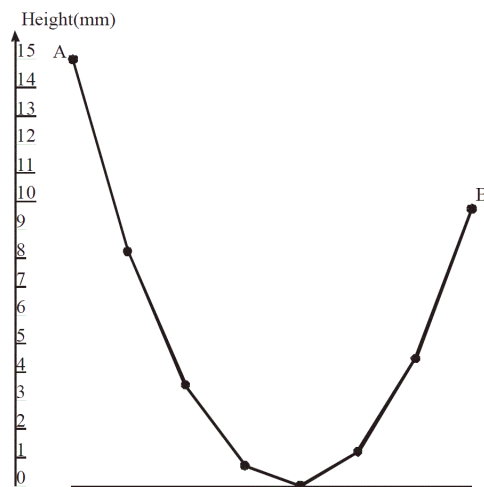
binsearch.in	binsearch.out
5	1 2
1 1 2 2 2	3 5
3	-1 -1
1 2 3	

## Задача D. Гирлянда (3 балла)

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

Гирлянда состоит из  $n$  лампочек на общем проводе. Один её конец закреплён на заданной высоте  $A$  мм ( $h_1 = A$ ). Благодаря силе тяжести гирлянда прогибается: высота каждой неконцевой лампы на 1 мм меньше, чем средняя высота ближайших соседей ( $h_i = \frac{(h_{i-1} + h_{i+1})}{2} - 1$  для  $1 < i < N$ ). Требуется найти минимальную высоту второго конца  $B$  ( $B = h_n$ ) при условии, что лишь одна лампочка может касаться земли, а для остальных выполняется условие  $h_i > 0$ .

Подсказка: используйте двоичный поиск.



### Формат входного файла

В первую строке входного файла содержится два числа  $n$  и  $A$  ( $3 \leq n \leq 1000$ ,  $n$  — целое,  $10 \leq A \leq 1000$ ,  $A$  — вещественное).

### Формат выходного файла

Вывести одно вещественное число  $B$  с двумя знаками после запятой.

### Пример

garland.in	garland.out
8 15	9.75
692 532.81	446113.34