

## Задача А. Двоичный поиск

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Реализуйте алгоритм бинарного поиска.

### Формат входных данных

В первой строке содержатся числа  $n$  и  $k$  ( $1 \leq n, k \leq 10^5$ ).

Во второй строке задаются  $n$  элементов первого массива, отсортированного по возрастанию, а в третьей строке —  $k$  элементов второго массива. Элементы массивов — целые числа, не превосходящие по модулю  $10^9$ .

### Формат выходных данных

Для каждого из  $k$  чисел второго массива выведите в отдельную строку «YES», если это число встречается в первом массиве, и «NO» в противном случае.

### Пример

стандартный ввод	стандартный вывод
10 5	NO
1 2 3 4 5 6 7 8 9 10	NO
-2 0 4 9 12	YES
	YES
	NO

## Задача В. Приближенный двоичный поиск

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Реализуйте алгоритм приближенного бинарного поиска.

### Формат входных данных

В первой строке входных данных содержатся числа  $n$  и  $k$  ( $1 \leq n, k \leq 10^5$ ). Во второй строке задаются  $n$  чисел первого массива, отсортированного по неубыванию, а в третьей строке –  $k$  чисел второго массива. Каждое число в обоих массивах по модулю не превосходит  $2 \cdot 10^9$ .

### Формат выходных данных

Для каждого из  $k$  чисел выведите в отдельную строку число из первого массива, наиболее близкое к данному. Если таких несколько, выведите меньшее из них.

### Пример

стандартный ввод	стандартный вывод
5 5	1
1 3 5 7 9	3
2 4 8 1 6	7
	1
	5

## Задача С. Массивы

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Даны два массива. Для каждого элемента второго массива определите, сколько раз он встречается в первом массиве.

### Формат входных данных

Первая строка входных данных содержит одно число  $n$  ( $1 \leq n \leq 10^5$ ) — количество элементов в первом массиве. Далее идет  $n$  целых чисел, не превосходящих по модулю  $10^9$  — элементы первого массива. Далее идет количество элементов  $m$  во втором массиве и  $m$  элементов второго массива с такими же ограничениями.

### Формат выходных данных

Выведите  $m$  чисел: для каждого элемента второго массива выведите, сколько раз такое значение встречается в первом массиве.

### Пример

стандартный ввод	стандартный вывод
3 1 2 1 4 0 1 2 3	0 2 1 0

## Задача D. Квадратный корень и квадратный квадрат

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Найдите такое число  $x$ , что  $x^2 + \sqrt{x} = C$ , с точностью не менее 6 знаков после точки.

### Формат входных данных

В единственной строке содержится вещественное число  $1 \leq C \leq 10^{10}$ .

### Формат выходных данных

Выведите одно число — искомый  $x$ .

### Примеры

стандартный ввод	стандартный вывод
2.0000000000	1.00000000000000000000
18.0000000000	4.00000000000000000000

## Задача Е. Корень кубического уравнения

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Дано кубическое уравнение  $ax^3 + bx^2 + cx + d = 0$  ( $a \neq 0$ ). Известно, что у этого уравнения ровно один корень. Требуется его найти.

### Формат входных данных

Во входных данных через пробел записаны четыре целых числа:  $-1000 \leq a, b, c, d \leq 1000$ .

### Формат выходных данных

Выведите единственный корень уравнения с точностью не менее 4 знаков после десятичной точки.

### Пример

стандартный ввод	стандартный вывод
1 -3 3 -1	1.0000005398739177931

## Задача F. Минимизируем максимум

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 3 секунды  
Ограничение по памяти: 512 мегабайт

Даны  $n$  нестрого возрастающих массивов  $A_i$  и  $m$  нестрого убывающих массивов  $B_j$ . Все массивы имеют одну и ту же длину  $l$ . Далее даны  $q$  запросов вида  $(i, j)$ , ответ на запрос – такое  $k$ , что  $\max(A_{ik}, B_{jk})$  минимален. Если таких  $k$  несколько, можно вернуть любое.

### Формат входных данных

На первой строке числа  $n, m, l$  ( $1 \leq n, m \leq 900; 1 \leq l \leq 3000$ ). Следующие  $n$  строк содержат описания массивов  $A_i$ . Каждый массив описывается перечислением  $l$  элементов. Элементы массива – целые числа от 0 до  $10^5 - 1$ . Далее число  $m$  и описание массивов  $B_j$  в таком же формате. Массивы и элементы внутри массива нумеруются с 1. На следующей строке число запросов  $q$  ( $1 \leq q \leq n \cdot m$ ). Следующие  $q$  строк содержат пары чисел  $i, j$  ( $1 \leq i \leq n, 1 \leq j \leq m$ ).

### Формат выходных данных

Выведите  $q$  чисел от 1 до  $l$  – ответы на запросы.

### Пример

стандартный ввод	стандартный вывод
4 3 5	3
1 2 3 4 5	4
1 1 1 1 1	3
0 99999 99999 99999 99999	5
0 0 0 0 99999	4
5 4 3 2 1	3
99999 99999 99999 0 0	1
99999 99999 0 0 0	2
12	2
1 1	4
1 2	4
1 3	3
2 1	
2 2	
2 3	
3 1	
3 2	
3 3	
4 1	
4 2	
4 3	

## Задача G. Коровы – в стойла

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

На прямой расположены стойла, в которые необходимо расставить коров так, чтобы минимальное расстояние между коровами было как можно больше.

### Формат входных данных

В первой строке вводятся числа  $N$  ( $2 < N \leq 10^5$ ) – количество стойл и  $K$  ( $1 < K < N$ ) – количество коров. Во второй строке задаются  $N$  натуральных чисел в порядке возрастания – координаты стойл (координаты не превосходят  $10^9$ ).

### Формат выходных данных

Выведите одно число – наибольшее возможное допустимое расстояние.

### Примеры

стандартный ввод	стандартный вывод
6 3 2 5 7 11 15 20	9
5 3 1 2 3 100 1000	99

## Задача Н. When democracy fails

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

В одной демократической стране приближаются парламентские выборы. Выборы проходят по следующей схеме: каждый житель страны, достигший восемнадцатилетнего возраста, отдает свой голос за одну из политических партий. После этого партия, которая набрала максимальное количество голосов, считается победившей на выборах и формирует правительство. Если несколько партий набрали одинаковое максимальное количество голосов, то они должны сформировать коалиционное правительство, что обычно приводит к длительным переговорам.

Один бизнесмен решил выгодно вложить свои средства и собрался поддержать на выборах некоторые партии. В результате поддержки он планирует добиться победы одной из этих партий, которая затем сформирует правительство, которое будет действовать в его интересах. При этом возможность формирования коалиционного правительства его не устраивает, поэтому он планирует добиться строгой победы одной из партий.

Чтобы повлиять на исход выборов, бизнесмен собирается выделить деньги на агитационную работу среди жителей страны. Исследование рынка показало, что для того, чтобы один житель сменил свои политические воззрения, требуется потратить одну условную единицу. Кроме того, чтобы  $i$ -я партия в случае победы сформировала правительство, которое будет действовать в интересах бизнесмена, необходимо дать лидеру этой партии взятку в размере  $p_i$  условных единиц. При этом некоторые партии оказались идеологически устойчивыми и не согласны на сотрудничество с бизнесменом ни за какие деньги.

По результатам последних опросов известно, сколько граждан планируют проголосовать за каждую партию перед началом агитационной компании. Помогите бизнесмену выбрать, какую партию следует подкупить, и какое количество граждан придется убедить сменить свои политические воззрения, чтобы выбранная партия победила, учитывая, что бизнесмен хочет потратить на всю операцию минимальное количество денег.

### Формат входных данных

В первой строке вводится целое число  $n$  – количество партий ( $1 \leq n \leq 10^5$ ). Следующие  $n$  строк описывают партии. Каждая из этих строк содержит по два целых числа:  $r_i$  – количество жителей, которые собираются проголосовать за эту партию перед началом агитационной компании, и  $b_i$  – взятка, которую необходимо дать лидеру партии для того, чтобы сформированное ей в случае победы правительство действовало в интересах бизнесмена ( $1 \leq r_i \leq 10^6$ ,  $1 \leq b_i \leq 10^6$  или  $b_i = -1$ ). Если партия является идеологически устойчивой, то  $b_i = -1$ . Гарантируется, что хотя бы одно  $b_i$  не равно  $-1$ .

### Формат выходных данных

В первой строке выведите минимальную сумму, которую придется потратить бизнесмену. Во второй строке выведите номер партии, лидеру которой следует дать взятку. В третьей строке выведите  $n$  целых чисел – количество голосов, которые будут отданы за каждую из партий после осуществления операции. Если оптимальных решений несколько, выведите любое.

### Примеры

стандартный ввод	стандартный вывод
3 7 -1 2 8 1 2	6 3 3 2 5
2 239 239 238 -1	239 1 239 238

## Задача I. Дремучий лес

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Чтобы помешать появлению СЭС в лагере, администрация ЛКШ перекопала единственную дорогу, соединяющую «Берендеевы поляны» с Судиславлем, теперь проехать по ней невозможно. Однако, трудности не остановили инспекцию, хотя для СЭС остается только одна возможность — дойти до лагеря пешком. Как известно, Судиславль находится в поле, а «Берендеевы поляны» — в лесу.

- Судиславль находится в точке с координатами  $(0, 1)$ .
- «Берендеевы поляны» находятся в точке с координатами  $(1, 0)$ .
- Граница между лесом и полем — горизонтальная прямая  $y = a$ , где  $a$  — некоторое число ( $0 < a < 1$ ).
- Скорость передвижения СЭС по полю составляет  $V_p$ , скорость передвижения по лесу —  $V_f$ . Вдоль границы можно двигаться как по лесу, так и по полю.

Администрация ЛКШ хочет узнать, сколько времени у нее осталось для подготовки к визиту СЭС. Она попросила вас выяснить, в какой точке инспекция СЭС должна войти в лес, чтобы дойти до «Берендеевых полей» как можно быстрее.

### Формат входных данных

В первой строке входного файла содержатся два положительных целых числа  $V_p$  и  $V_f$  ( $1 \leq V_p, V_f \leq 10^5$ ). Во второй строке содержится единственное вещественное число — координата по оси  $O_y$  границы между лесом и полем  $a$  ( $0 < a < 1$ )

### Формат выходных данных

В единственной строке выходного файла выведите вещественное число с точностью не менее 6 знаков после запятой — координата по оси  $O_x$  точки, в которой инспекция СЭС должна войти в лес.

### Примеры

стандартный ввод	стандартный вывод
5 3 0.4	0.7833106
5 5 0.5	0.5000000

## Задача J. Взвешенно минимизируем максимум

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Вам дано  $n$  точек на плоскости:  $A_1, A_2, \dots, A_n$ . У  $i$ -й точки есть вес  $w_i$ . Найдите такую точку  $B$ , что значение максимума взвешенного расстояния  $\max_{i=1}^n w_i \cdot |A_i B|$  минимально.

### Формат входных данных

Во вводе содержится один или несколько тестовых случаев.

В первой строке каждого тестового случая задано число  $n$  ( $1 \leq n \leq 100$ ) — количество точек. В каждой из следующих  $n$  строк задано по три числа:  $x_i$ ,  $y_i$  и  $w_i$ . Все числа целые и не превосходят  $10^7$  по модулю. Вес каждой точки строго положителен.

Тестовые случаи следуют друг за другом без каких-либо пропусков. Ввод заканчивается строкой, содержащей одно число 0. Его не надо считать ещё одним тестовым случаем. Сумма чисел  $n$  во всех тестовых случаях не превосходит 100.

### Формат выходных данных

В ответ на каждый тестовый случай выведите два числа — координаты точки  $B$ . Ваш ответ будет считаться правильным, если абсолютная или относительная погрешность максимума взвешенного расстояния не будет превосходить  $10^{-9}$ .

### Пример

стандартный ввод	стандартный вывод
2	1.0 1.0
2 2 1	2.4 3.6
0 0 1	
3	
0 0 1	
6 0 2	
0 6 3	
0	