

## Задача А. Приближенный двоичный поиск

Имя входного файла: `stdin`  
Имя выходного файла: `stdout`  
Ограничение по времени: 6 секунды  
Ограничение по памяти: 64 мегабайта

### Формат входных данных

В первой строке входных данных содержатся числа  $N$  и  $K$  ( $1 \leq N, K \leq 100\,000$ ). Во второй строке задаются  $N$  чисел первого массива, отсортированного по возрастанию, а в третьей строке —  $K$  чисел второго массива. Каждое число в обоих массивах по модулю не превосходит  $10^9$ .

### 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.

### Примеры

stdin	stdout
5 5	1
1 3 5 7 9	3
2 4 8 1 6	7
	1
	5

### Замечание

В этой задаче нельзя пользоваться встроенной сортировкой.

## Задача В. Квадратный корень и квадратный квадрат

Имя входного файла: `stdin`  
Имя выходного файла: `stdout`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

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  $1.0 \leq C \leq 10^{10}$ .

### Формат выходных данных

Выведите одно число — искомый  $x$ .

### Примеры

stdin	stdout
2.0000000000	1.0000000000
18.0000000000	4.0000000000

## Задача С. Шарики

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Организаторы детского праздника планируют надуть для него  $M$  воздушных шариков. С этой целью они пригласили  $N$  добровольных помощников,  $i$ -й среди которых надувает шарик за  $T_i$  минут, однако каждый раз после надувания  $Z_i$  шариков устаёт и отдыхает  $Y_i$  минут. Теперь организаторы праздника хотят узнать, через какое время будут надуты все шарики при наиболее оптимальной работе помощников, и сколько шариков надуёт каждый из них. (Если помощник надул шарик, и должен отдохнуть, но больше шариков ему надувать не придётся, то считается, что он закончил работу сразу после окончания надувания последнего шарика, а не после отдыха).

### Формат входных данных

В первой строке входных данных задаются числа  $M$  и  $N$  ( $0 \leq M \leq 15\,000$ ,  $1 \leq N \leq 1000$ ). Следующие  $N$  строк содержат по три целых числа —  $T_i$ ,  $Z_i$  и  $Y_i$  соответственно ( $1 \leq T_i \leq 100$ ,  $1 \leq Y_i \leq 100$ ,  $1 \leq Z_i \leq 1'\,000$ ).

### Формат выходных данных

В первой строке программа должна вывести значение  $T$  — минимальное время, за которое будут надуты все шарики. Во второй строке выведите  $N$  чисел — количество шариков, надутых каждым из приглашённых помощников. Разделяйте числа пробелами. Если распределений шариков несколько, выведите любое из них. Необходимо надуть ровно  $N$  шариков, нельзя надуть большее число шариков, даже если это занимает такое же время.

### Примеры

стандартный ввод	стандартный вывод
10 3 1 2 3 3 10 3 2 4 3	8 4 2 4
1 3 1 1 100 2 1 100 3 1 100	1 1 0 0
1 2 1 1 1 1 1 1	1 1 0

## Задача D. Дремучий лес

Имя входного файла: `stdin`  
Имя выходного файла: `stdout`  
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 остаётся 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  $(0 \leq a \leq 1)$ .
- 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 неё осталось для подготовки к визиту СЭС. Она попросила вас выяснить, в какой точке инспекция СЭС должна войти в лес, чтобы дойти до «Берендеевых полян» 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  $Oy$  границы между лесом и полем  $a$  ( $0 \leq a \leq 1$ )

### Формат выходных данных

В единственной строке выходного файла выведите вещественное число с точностью не менее 7 знаков после запятой — координата по оси  $Ox$  точки, в которой инспекция СЭС должна войти в лес.

### Примеры

stdin	stdout
5 3 0.4	0.783310604
5 5 0.5	0.500000000

## Задача Е. Объединение отрезков

Имя входного файла: `stdin`  
Имя выходного файла: `stdout`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Решая задачу из контрольной по математике, Вася получил ответ в виде объединения  $N$  отрезков  $[L_i, R_i]$  на числовой прямой. Однако, некоторые из этих отрезков могут пересекаться друг с другом, что не слишком нравится Васе. Ваша задача — представить Васин ответ в виде объединения минимального количества отрезков.

### Формат входных данных

В первой строке указано число  $N$  ( $1 \leq N \leq 50000$ ). В следующих  $N$  строках перечислены пары целых чисел  $L_i$  и  $R_i$  ( $|L_i|, |R_i| \leq 50000$ ), каждая пара с новой строки, числа в парах отделены друг от друга одним или несколькими пробелами.

### Формат выходных данных

В первой строке выведите число  $M$  — количество отрезков в искомом объединении. В следующих  $M$  строках выведите сами эти отрезки в том же формате, что и во входном файле. Список отрезков необходимо упорядочить по возрастанию левого конца.

### Примеры

stdin	stdout
4	2
0 2	0 3
4 5	4 6
1 3	
5 6	

## Задача F. Точки и отрезки [геометрия HARD]

Имя входного файла: `stdin`  
Имя выходного файла: `stdout`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Дано  $n$  отрезков на числовой прямой и  $m$  точек на этой же прямой. Для каждой из данных точек определите, скольким отрезкам она принадлежит. Точка  $x$  считается принадлежащей отрезку с концами  $a$  и  $b$ , если выполняется двойное неравенство  $\min(a, b) \leq x \leq \max(a, b)$ .

### Формат входных данных

Первая строка содержит два целых числа  $n$  ( $1 \leq n \leq 50000$ ) — число отрезков и  $m$  ( $1 \leq m \leq 50000$ ) — число точек. В следующих  $n$  строках по два целых числа  $a_i$  и  $b_i$  — координаты концов соответствующего отрезка. В последней строке  $m$  целых чисел — координаты точек. Все числа во входном файле не превосходят по модулю  $10^9$ .

### Формат выходных данных

В выходной файл выведите  $m$  чисел — для каждой точки количество отрезков, в которых она содержится.

### Примеры

stdin	stdout
3 2 0 5 -3 2 7 10 1 6	2 0
1 3 10 -10 -100 100 0	0 0 1

## Задача G. Кассы

Имя входного файла: `stdin`  
Имя выходного файла: `stdout`  
Ограничение по времени: 0.5 секунда  
Ограничение по памяти: 64 мегабайта

На одном из московских вокзалов билеты продают  $N$  касс. Каждая касса работает без перерыва определенный промежуток времени по фиксированному расписанию (одному и тому же каждый день). Требуется определить, на протяжении какого времени в течение суток работают все кассы одновременно.

### Формат входных данных

Сначала вводится одно целое число  $N$  ( $0 < N \leq 10000$ ).

В каждой из следующих  $N$  строк через пробел расположены 6 целых чисел, первые три из которых обозначают время открытия кассы в часах, минутах и секундах (часы — целое число от 0 до 23, минуты и секунды — целые числа от 0 до 59), оставшиеся три — время закрытия в том же формате. Числа разделены пробелами.

Время открытия означает, что в соответствующую ему секунду касса уже работает, а время закрытия — что в соответствующую секунду касса уже не работает. Например, касса, открытая с 10 ч 30 мин 30 с до 10 ч 35 мин 30 с, ежедневно работает 300 секунд.

Если время открытия совпадает с временем закрытия, то касса работает круглосуточно. Если первое время больше второго, то касса начинает работу до полуночи, а заканчивает — на следующий день.

### Формат выходных данных

Требуется вывести одно число — суммарное время за сутки (в секундах), на протяжении которого работают все  $N$  касс.

### Примеры

stdin	stdout
3 1 0 0 23 0 0 12 0 0 12 0 0 22 0 0 2 0 0	7200
2 9 30 0 14 0 0 14 15 0 21 0 0	0
2 14 0 0 18 0 0 10 0 0 14 0 1	1