

## Задача А. Простая задача

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.5 секунд  
Ограничение по памяти: 256 мегабайт

Найдите оптимальное решение задачи линейного программирования.

$$\begin{cases} a_{11}x_1 + \dots + a_{1n}x_n \leq b_1 \\ a_{21}x_1 + \dots + a_{2n}x_n \leq b_2 \\ \dots \\ a_{m1}x_1 + \dots + a_{mn}x_n \leq b_m \\ x_1 \geq 0 \\ \dots \\ x_n \geq 0 \\ c_1x_1 + \dots + c_nx_n \rightarrow \max \end{cases}$$

### Формат входных данных

Первая строка входного файла содержит два целых числа:  $n$  и  $m$  — количество переменных и количество уравнений ( $1 \leq n, m \leq 5$ ). Следующие  $m$  строк содержат по  $n + 1$  целому числу:  $a_{i1}, \dots, a_{in}, b_i$ . Следующая строка содержит  $n$  целых чисел:  $c_1, \dots, c_n$ . Все числа во входном файле не превышают 100 по модулю.

### Формат выходных данных

Выведите одно число: максимальное значение  $c_1x_1 + \dots + c_nx_n$ .

Гарантируется, что решение существует и максимальное значение достигается.

### Примеры

стандартный ввод	стандартный вывод
2 2 1 2 3 2 1 3 1 1	2.000000000
1 1 1 2 -2	-0.000000000
1 1 4 5 3	3.750000000

## Задача В. Простая задача 2

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.5 секунд  
Ограничение по памяти: 256 мегабайт

Найдите оптимальное решение задачи линейного программирования.

$$\begin{cases} a_{11}x_1 + \dots + a_{1n}x_n \leq b_1 \\ a_{21}x_1 + \dots + a_{2n}x_n \leq b_2 \\ \dots \\ a_{m1}x_1 + \dots + a_{mn}x_n \leq b_m \\ x_1 \geq 0 \\ \dots \\ x_n \geq 0 \\ c_1x_1 + \dots + c_nx_n \rightarrow \max \end{cases}$$

### Формат входных данных

Первая строка входного файла содержит два целых числа:  $n$  и  $m$  — количество переменных и количество уравнений ( $1 \leq n, m \leq 5$ ). Следующие  $m$  строк содержат по  $n + 1$  целому числу:  $a_{i1}, \dots, a_{in}, b_i$ . Следующая строка содержит  $n$  целых чисел:  $c_1, \dots, c_n$ . Все числа во входном файле не превышают 100 по модулю.

### Формат выходных данных

Выведите одно число: максимальное значение  $c_1x_1 + \dots + c_nx_n$ . Если решения нет, выведите “No solution”. Если можно получить сколь угодно большое значение, выведите “Unbounded”.

### Примеры

стандартный ввод	стандартный вывод
2 2 1 2 3 2 1 3 1 1	2.0000000000000000
2 1 -1 -1 0 1 1	Unbounded
2 1 1 1 -1 1 1	No solution

## Задача С. Простая задача с хорошими тестами

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.5 секунд  
Ограничение по памяти: 256 мегабайт

Найдите оптимальное решение задачи линейного программирования.

$$\begin{cases} a_{11}x_1 + \dots + a_{1n}x_n \leq b_1 \\ a_{21}x_1 + \dots + a_{2n}x_n \leq b_2 \\ \dots \\ a_{m1}x_1 + \dots + a_{mn}x_n \leq b_m \\ x_1 \geq 0 \\ \dots \\ x_n \geq 0 \\ c_1x_1 + \dots + c_nx_n \rightarrow \max \end{cases}$$

### Формат входных данных

Входной файл содержит один или несколько тестов. На первой строке число тестов, далее сами тесты. Каждый тест описывается следующим образом. Первая строка теста содержит два целых числа:  $n$  и  $m$  — количество переменных и количество неравенств ( $1 \leq n, m \leq 5$ ). Следующие  $m$  строк содержат по  $n + 1$  целому числу:  $a_{i1}, \dots, a_{in}, b_i$ . Следующая строка содержит  $n$  целых чисел:  $c_1, \dots, c_n$ . Все числа во входном файле целые и не превышают 100 по модулю.

### Формат выходных данных

Для каждого теста выведите одну строку.

Выведите максимальное значение  $c_1x_1 + \dots + c_nx_n$ , пробел, двоеточие, пробел, сами числа  $x_1x_2\dots x_n$ . Все числа выводите с максимальной точностью. Если решения нет, выведите “No solution”. Если можно получить сколь угодно большое значение, выведите “Unbounded”.

### Пример

стандартный ввод	стандартный вывод
4	2.000000000 : 1.000000000 1.000000000
2 2	Unbounded
1 2 3	No solution
2 1 3	1.500000000 : 0.000000000 1.500000000
1 1	
2 1	
-1 -1 0	
1 1	
2 1	
1 1 -1	
1 1	
2 2	
1 2 3	
2 1 3	
-10 1	

## Задача D. Road times

Имя входного файла: стандартный ввод  
 Имя выходного файла: стандартный вывод  
 Ограничение по времени: 0.5 секунд  
 Ограничение по памяти: 256 мегабайт

Дорожная сеть страны – ориентированный граф. У каждого ребра есть длина, целое число от 1 до 1000, и ограничение на максимальную скорость, вещественное число от 30 до 60, в километрах в час. Длины вам известны, а ограничения – нет. Известно, что когда водитель такси берётся доставить пассажира из вершины  $a$  в вершину  $b$ , он осуществляет перевозку строго по кратчайшему пути между вершинами и едет с максимальной допустимой скоростью. Длина пути – сумма длин рёбер. Также известно, что  $\forall a, b \exists$  единственный кратчайший путь из  $a$  в  $b$ .

Ваша задача – по длинам рёбер и уже сделанным поездкам  $a_i b_i time_i$  оценить минимальное и максимальное время в пути между вершинами  $c_j d_j$

### Формат входных данных

На первой строке число вершин  $n$  ( $1 \leq n \leq 30$ ).

Вершины нумеруются числами от 0 до  $n - 1$ .

Следующие  $n$  строк содержат матрицу  $n \times n$  длин дорог. Дороги односторонние.

Отсутствие дорог обозначено числом  $-1$ , длины дорог целые от 1 до 1000.

Всего не более 100 дорог.

Далее идёт число  $r$  ( $1 \leq r \leq 100$ ) и  $r$  уже известных маршрутов  $a_i b_i time_i$ ,  $time_i \in \mathbb{R}$ .

Времена в минутах (не в часах).

Затем число запросов  $q$  ( $1 \leq q \leq 100$ ) и  $q$  строк  $c_j d_j$ .

### Формат выходных данных

Для каждого запроса выведите четыре числа – откуда, куда, min время, max время.

### Пример

стандартный ввод	стандартный вывод
3	0 1 50.0000000000000000 80.0000000000000000
0 50 -1	1 2 40.0000000000000000 69.9999999999999986
55 0 40	1 0 55.0000000000000000 110.0000000000000000
-1 40 0	
1	
0 2 120	
3	
0 1	
1 2	
1 0	

## Задача Е. Гиперраздление

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.5 секунд  
Ограничение по памяти: 256 мегабайт

Вам даны  $N$   $d$ -мерных точек. Нужно разделить их  $d - 1$  мерной плоскостью, проходящей через точку  $(0, \dots, 0)$ . Гарантируется, что решение всегда существует.

### Формат входных данных

В первой строке  $N$  ( $1 \leq N \leq 5000$ ),  $d$  ( $1 \leq d \leq 10$ ). Далее  $N$  строк содержат по  $d + 1$  чисел. Первые  $d$  чисел — вещественные числа, координаты очередной точки  $x_{j,1}, x_{j,2}, \dots, x_{j,d}$ . Последнее число — целое число  $y_j$  ( $|y_j| = 1$ ). Число  $y_j$  определяет положение точки относительно плоскости, читайте дальше для более точного понимания.

Координаты точек по модулю не превосходят 100.0. Все вещественные числа содержат не более 6 знаков после десятичной точки.

### Формат выходных данных

Выведите  $d$  вещественных чисел  $n_1, n_2, \dots, n_d$  — нормаль плоскости. Нормаль — вектор произвольной **положительной** длины. Для каждой точки  $\text{sign}(\sum_{i=1}^d x_{j,i} \cdot n_i)$  должен быть равен ее  $y_j$ . Если решений несколько, можно вывести любое.

Ответ будет считаться корректным, если для каждой точки  $|\sum_{i=1}^d x_{j,i} \cdot n_i| \geq 10^{-4}|n|$ , где  $|n|$  — длина вектора нормали.

### Примеры

стандартный ввод	стандартный вывод
2 1 1 -1 -1 1	-1.000000000
4 3 -99.749 12.71 -61.33 1 61.7 17.00 -4.0 -1 -29.94 79.192 64.56 1 49.320 -65.178 71.788 -1	-0.891651465 0.415648308 -0.179427280

## Задача F. Триатлон

Имя входного файла: `tri.in`  
Имя выходного файла: `tri.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Триатлон — атлетическое соревнование, состоящее из трех последовательных этапов, которые суммарно должны быть пройдены как можно быстрее. Первый этап — плавание, второй — велогонка, а третий — бег.

Известна скорость каждого участника для каждого этапа. Жюри соревнования может выбрать длину каждого этапа любым образом так, чтобы каждый этап не был нулевой длины. В результате, иногда жюри может выбрать длины этапов так, чтобы определенный участник стал победителем соревнования.

### Формат входных данных

В первой строке входного файла задано целое число  $N$  — количество участников соревнования ( $1 \leq N \leq 100$ ). Следующие  $N$  строк содержат по 3 целых числа  $V, U$  и  $W$  ( $1 \leq U, V, W \leq 10000$ ), определяющие скорость очередного участника на этапах соревнования.

### Формат выходных данных

Для каждого участника выведите одну строку, содержащую слово *Yes* в случае если жюри может подобрать длины этапов так, чтобы этот участник стал победителем (т.е. этот участник — единственный, кто первый дошел до финиша), или слово *No* в противном случае.

### Примеры

<code>tri.in</code>	<code>tri.out</code>
9	Yes
10 2 6	Yes
10 7 3	Yes
5 6 7	No
3 2 7	No
6 2 6	No
3 5 7	Yes
8 4 6	No
10 4 2	Yes
1 8 7	