

Содержание

Must have	2
Задача 31A. Простая задача [1, 256]	2
Задачи здорового	3
Задача 31B. Простая задача [1, 256]	3
Задача 31C. Простая задача с хорошими тестами [1, 256]	4
Для искателей острых ощущений	5
Задача 31D. Массивы-палиндромы [1, 256]	5
Для мастеров AI	6
Задача 31E. Road times [1, 256]	7

Обратите внимание, входные данные лежат в **стандартном потоке ввода** (он же stdin), вывести ответ нужно в **стандартный поток вывода** (он же stdout).

Обратите внимание на GNU C++ компиляторы с суффиксом inc.

Подни можно пользоваться **дополнительной библиотекой** (optimization.h).

То есть, использовать быстрый ввод-вывод: **пример про числа и строки**.

И быструю аллокацию памяти (ускоряет vector-set-map-весь-STL): **пример**.

Для тех, кто хочет разобраться, как всё это работает.

Короткая версия быстрого ввода-вывода (**тык**) и короткая версия аллокатора (**тык**).

Must have

Задача 31А. Простая задача [1, 256]

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.

Пример

stdin	stdout
2 2 1 2 3 2 1 3 1 1	2.0
1 1 1 2 -2	0.0
1 1 4 5 3	3.75

Подсказка по решению

Гаусс. Много раз.

Задачи здорового

Задача 31В. Простая задача [1, 256]

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”.

Пример

stdin	stdout
2 2 1 2 3 2 1 3 1 1	2.0
2 1 -1 -1 0 1 1	Unbounded
2 1 1 1 -1 1 1	No solution

Подсказка по решению

Гаусс. Много раз.

Задача 31С. Простая задача с хорошими тестами [1, 256]

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

stdin	stdout
4	2.0 : 1.0 1.0
2 2	Unbounded
1 2 3	No solution
2 1 3	1.5 : 0 1.5
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	

Подсказка по решению

Гаусс. Очень много раз. И повышенная акууратность.

Для искателей острых ощущений

Задача 31D. Массивы-палиндромы [1, 256]

Даны n целых чисел от 1 до 1 000. Числа могут повторяться. В сколько из $2^n - 1$ непустых подмножеств этих n чисел произведение является точным квадратом?

Формат входных данных

На первой строке n ($3 \leq n \leq 1\,000$).

На второй строке n целых чисел от 1 до 1 000.

Формат выходных данных

Выведите одно число – количество непустых подмножеств по модулю $10^9 + 7$.

Пример

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

Подсказка по решению

Гаусс и $\mathbb{F}_2$.

Для мастеров AI

Правила.

Это блок задач про промтинг. Пользоваться можно только бесплатными версиями ИИ. В шапке исходника указывать последовательность промтов, и сайты, куда они были отправлены. Если в процессе использования ИИ вы получили какие-то важные идеи для решения, это тоже часть решения. Если с помощью ИИ вы написали генератор тестов или стресс-тест, это тоже следует указывать. ЗадOCUMENTИРУЙТЕ, пожалуйста, сделанную работу в шапке отосланного решения. Если это нужно, вы можете писать часть кода сами.

Задача 31Е. Road times [1, 256]

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.

Примеры

stdin	stdout
3	0 1 50.0 80.0
0 50 -1	1 2 40.0 70.0
55 0 40	1 0 55.0 110.0
-1 40 0	
1	
0 2 120	
3	
0 1	
1 2	
1 0	

Подсказка по решению

Не все способы решать LP одинаково полезны.

Можно случайно получить слишком медленное решение.