

## Задача А. Максимальный поток минимальной стоимости

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

Задан ориентированный граф, каждое ребро которого обладает пропускной способностью и стоимостью. Найдите максимальный поток минимальной стоимости из вершины с номером 1 в вершину с номером  $n$ .

### Формат входных данных

Первая строка входного файла содержит  $n$  и  $m$  — количество вершин и количество ребер графа ( $2 \leq n \leq 100$ ,  $0 \leq m \leq 1000$ ). Следующие  $m$  строк содержат по четыре целых числа: номера вершин, которые соединяет соответствующее ребро графа, его пропускную способность и его стоимость. Пропускные способности и стоимости не превосходят  $10^5$ .

### Формат выходных данных

В выходной файл выведите одно число — цену максимального потока минимальной стоимости из вершины с номером 1 в вершину с номером  $n$ . Ответ не превышает  $2^{63} - 1$ . Гарантируется, что в графе нет циклов отрицательной стоимости.

### Примеры

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

### Замечание

В этой задаче достаточно несколько раз пустить Форд-Беллмана...

## Задача В. Автоматное программирование

Имя входного файла: `schedule.in`  
Имя выходного файла: `schedule.out`  
Ограничение по времени: 5 секунд  
Ограничение по памяти: 256 мегабайт

В один замечательный день в компанию «Х» завезли  $k$  автоматов. И не простых автоматов, а автоматов-программистов! Это был последний неудачный шаг перед переходом на андроидов-программистов, но это уже совсем другая история.

В компании сейчас  $n$  задач, для каждой из которых известно время начала ее выполнения  $s_i$ , длительность ее выполнения  $t_i$  и прибыль компании от ее завершения  $c_i$ . Любой автомат может выполнять любую задачу, ровно одну в один момент времени. Если автомат начал выполнять задачу, то он занят все моменты времени с  $s_i$  по  $s_i + t_i - 1$  включительно и не может переключиться на другую задачу.

Вам требуется выбрать набор задач, которые можно выполнить с помощью этих  $k$  автоматов и который принесет максимальную суммарную прибыль.

### Формат входных данных

В первой строке записаны два целых числа  $n$  и  $k$  ( $1 \leq n \leq 1000$ ,  $1 \leq k \leq 50$ ) — количество задач и количество автоматов, соответственно.

В следующих  $n$  строках через пробелы записаны тройки целых чисел  $s_i, t_i, c_i$  ( $1 \leq s_i, t_i \leq 10^9$ ,  $1 \leq c_i \leq 10^6$ ),  $s_i$  — время начала выполнения  $i$ -го задания,  $t_i$  — длительность  $i$ -го задания, а  $c_i$  — прибыль от его выполнения.

### Формат выходных данных

Выведите  $n$  целых чисел  $x_1, x_2, \dots, x_n$ . Число  $x_i$  должно быть равно 1, если задачу  $i$  следует выполнить, и 0 в противном случае.

Если оптимальных решений несколько, то выведите любое из них.

### Примеры

<code>schedule.in</code>	<code>schedule.out</code>
3 1 2 7 5 1 3 3 4 1 3	0 1 1
5 2 1 5 4 1 4 5 1 3 2 4 1 2 5 6 1	1 1 0 0 1

## Задача С. Аарельские горы

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

Вы — могучий волшебник, путешествующий по Аарельскому королевству. Ваш путь лежит через крутые холмы и горы. Горная цепочка может быть представлена, как последовательность  $n$  регионов; каждый регион имеет свою *высоту*  $h_i$ .

К сожалению, у вас нет никакого скалолазного оборудования (и, честно говоря, никакого умения его использовать), и ваши заклинания полета требуют перезарядки. Однако, вы сможете пересечь эту область, если высоты регионов будут образовывать неубывающую последовательность, то есть  $h_i \leq h_{i+1}$  для всех  $i$  от 1 до  $(n - 1)$  (вы путешествуете справа налево, и падения с высокой высоты для вас опасности не представляют).

Единственное заклинание, которое может вам помочь, это заклинание Землетрясение. Это заклинание влияет на несколько соседних регионов и позволяет вам изменить их высоты. Вы знаете  $m$  видов Землетрясения; каждое заклинание имеет свою длину  $l_i$  (количество соседних регионов, затронутых этим заклинанием), свою энергетическую стоимость  $c_i$  и либо увеличивает, либо уменьшает высоты всех затронутых регионов (одно заклинание может либо повышать, либо понижать регионы, но не и то, и то). Каждое заклинание может быть применено к любому отрезку длины  $l_i$ , полностью лежащему внутри области из  $n$  регионов, и может быть применено сколько угодно раз (но его энергетическую стоимость придется платить при каждом применении). Разрешено, чтобы какие-то регионы имеют отрицательную высоту после применения заклинания.

Определите, возможно ли изменить ландшафт так, что вы сможете пересечь заданную область, и, если это возможно, найдите минимальную суммарную энергию, которую придется потратить на применение заклинаний.

### Формат входных данных

Первая строка содержит два числа  $n$  и  $m$ , разделенных пробелами ( $1 \leq n, m \leq 200$ ).

Вторая строка содержит  $n$  чисел  $h_i$  — изначальную высоту регионов ( $0 \leq h_i \leq 10^6$ ).

Следующих  $m$  строчек описывают виды заклинаний Землетрясение, которые вы знаете,  $i$ -я из этих строк содержит описание заклинания в формате " $t_i l_i c_i$ "; где  $t_i$  это символ "+" если заклинание позволяет увеличить высоту на 1, или "-" если оно позволяет уменьшить высоту на 1;  $l_i$  и  $c_i$  — длина и энергетическая стоимость заклинания, соответственно ( $1 \leq l_i \leq n, 1 \leq c_i \leq 10^6$ ).

### Формат выходных данных

Если вы можете пройти через эту область, выведите минимальную возможную суммарную энергию, потраченную на заклинания, иначе, выведите  $-1$ .

### Примеры

стандартный ввод	стандартный вывод
3 2 3 2 1 + 1 1 - 1 1	2
3 1 3 2 1 + 2 1	-1

## Задача D. Цикл минимального среднего веса

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 512 мегабайт

Дан ориентированный граф, содержащий  $n$  вершин и  $m$  ребер. Каждое ребро имеет некоторый вес. Простым циклом называется последовательность рёбер  $(a_1, b_1), (a_2, b_2), \dots, (a_k, b_k)$  такая, что все  $a_i$  различны,  $b_i = a_{i+1}$ ,  $b_k = a_1$ . Весом цикла называется сумма весов составляющих его ребер. Средним весом цикла называется отношение веса цикла к числу составляющих его ребер.

Требуется найти в графе **простой** цикл минимального среднего веса.

### Формат входных данных

Первая строка входного файла содержит числа  $n$  и  $m$  ( $3 \leq n \leq 1000$ ,  $3 \leq m \leq 2000$ ). Следующие  $m$  строк описывают ребра графа. Каждое ребро описывается тремя целыми числами  $u_i, v_i$  и  $w_i$ : номер начальной вершины, номер конечной вершины и вес ребра ( $1 \leq u_i, v_i \leq n$ ,  $-1000 \leq w_i \leq 1000$ ). Гарантируется, что в графе есть цикл. В графе может быть несколько ребер между одной и той же парой вершин, а также могут быть петли.

### Формат выходных данных

На первой строке выведите число  $z$  — минимальный возможный средний вес цикла в заданном графе. На второй строке выведите число  $k$  — число ребер в таком цикле. На третьей строке выведите  $k$  чисел — номера ребер в порядке обхода вдоль цикла. Ребра пронумерованы от 1 до  $m$  в порядке, в котором они заданы во входном файле. Если возможных циклов несколько, можно вывести любой. Требуется вывести простой цикл. Число  $z$  должно быть выведено с абсолютной или относительной погрешностью  $10^{-9}$ .

### Пример

стандартный ввод	стандартный вывод
5 8	2.500000000000000000
1 2 10	4
2 3 1	2 3 4 7
3 4 2	
4 5 3	
5 1 8	
5 5 7	
5 2 4	
3 5 4	

## Задача E. Minimum cost b-flow

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	10 секунд
Ограничение по памяти:	512 мегабайт

You are given a graph with  $n$  vertices and  $m$  edges, amount of supply/demand associated with each vertex, and flow lower/upper bound associated with each edge. The  $v$ -th vertex has  $b_v$  amount of supply if  $b_v$  was positive, and  $-b_v$  amount of demand otherwise. The  $e$ -th edge is directed from vertex  $s_e$  to  $t_e$  and has flow lower bound  $l_e$ , flow upper bound  $u_e$ , and cost per a unit of flow  $c_e$ .

Your task is to find and print the minimum cost **b**-flow value  $z$ , the flow **f** and its dual **p** that achieves the minimum, that is,  $z$ ,  $\mathbf{f} = \{f_e\}_{e=0\dots m-1}$ , and  $\mathbf{p} = \{p_v\}_{v=0\dots n-1}$  that satisfies following constraints;

- $z = \sum_e c_e f_e$
- $l_e \leq f_e \leq u_e$  (Capacity constraints)
- $\sum_{e \in \delta^+(v)} f_e - \sum_{e \in \delta^-(v)} f_e = b_v$  (Flow conservation constraints)
- $f_e > l_e \Rightarrow c_e + p_{s_e} - p_{t_e} \leq 0$  (Complementary slackness conditions)
- $f_e < u_e \Rightarrow c_e + p_{s_e} - p_{t_e} \geq 0$  (Complementary slackness conditions)
- $|p_v| \leq 10^{15}$

where  $\delta^+(v)$  is the set of edges leaving vertex  $v$  and  $\delta^-(v)$  is the set of edges entering vertex  $v$ , i.e.  $\delta^+(v) = \{e | s_e = v\}$  and  $\delta^-(v) = \{e | t_e = v\}$ .

If there's no such values, output "infeasible" instead.

- $0 \leq n \leq 100$
- $0 \leq m \leq 1000$
- $0 \leq s_e < n$
- $0 \leq t_e < n$
- $|b_v| \leq 10^9$
- $|l_e| \leq 10^9$
- $|u_e| \leq 10^9$
- $|c_e| \leq 10^9$
- $l_e \leq u_e$

All of the values are integral

Note that

the input graph may contain self loops, and the result value may exceed  $2^{64}$ .

### Формат входных данных

The first line contains numbers  $n$  and  $m$ .

The next  $n$  lines contain numbers  $b_i$ .

Next  $m$  lines contain numbers  $s_i$   $t_i$   $l_i$   $u_i$   $c_i$ .

## Примеры

стандартный ввод	стандартный вывод
3 5 1 -1 0 0 1 1 2 1 1 2 0 2 2 2 0 -3 5 1 0 2 0 3 -2 2 1 0 1 0	-2 0 -1 -1 1 0 3 3 0
2 1 -1 1 0 0 -1 1 0	infeasible
2 1 1 0 0 1 -10 10 0	infeasible