

## Содержание

<b>Must have</b>	<b>2</b>
<b>Задача 5A. Максимальный поток минимальной стоимости [0.2 sec, 256 mb]</b>	<b>2</b>
<b>Обязательные задачи</b>	<b>3</b>
<b>Задача 5B. В поисках невест [0.2 sec, 256 mb]</b>	<b>3</b>
<b>Задача 5C. <math>k</math> паросочетаний [0.5 sec, 256 mb]</b>	<b>4</b>
<b>Задача 5D. Автоматное программирование [0.3 sec, 256 mb]</b>	<b>5</b>
<b>Дополнительные задачи</b>	<b>6</b>
<b>Задача 5E. Глобальный разрез [1.5 sec, 256 mb]</b>	<b>6</b>
<b>Задача 5F. Roads [1 sec, 256 mb]</b>	<b>7</b>

---

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

В некоторых задачах большой ввод и вывод. Пользуйтесь **быстрым вводом-выводом**.

В некоторых задачах нужен STL, который активно использует динамическую память (set-ы, map-ы) **переопределение стандартного аллокатора** ускорит вашу программу.

Обратите внимание на GNU C++ компиляторы с суффиксом inc, они позволяют пользоваться **дополнительной библиотекой**. Под ними можно сдать **вот это**.

## Must have

### Задача 5А. Максимальный поток минимальной стоимости [0.2 sec, 256 mb]

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

stdin	stdout
4 5 1 2 1 2 1 3 2 2 3 2 1 1 2 4 2 1 3 4 2 3	12

#### Подсказка по решению

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

## Обязательные задачи

### Задача 5В. В поисках невест [0.2 сек, 256 mb]

Однажды король Флатландии решил отправить  $k$  своих сыновей на поиски невест. Всем известно, что во Флатландии  $n$  городов, некоторые из которых соединены дорогами. Король живет в столице, которая имеет номер 1, а город с номером  $n$  знаменит своими невестами.

Итак, король повелел, чтобы каждый из его сыновей добрался по дорогам из города 1 в город  $n$ . Поскольку, несмотря на обилие невест в городе  $n$ , красивых среди них не так много, сыновья опасаются друг друга. Поэтому они хотят добраться до цели таким образом, чтобы никакие два сына не проходили по одной и той же дороге (даже в разное время). Так как король любит своих сыновей, он хочет, чтобы среднее время сына в пути до города назначения было минимально.

#### Формат входных данных

В первой строке входного файла находятся числа  $n$ ,  $m$  и  $k$  — количество городов и дорог во Флатландии и сыновей короля, соответственно ( $2 \leq n \leq 200$ ,  $1 \leq m \leq 2000$ ,  $1 \leq k \leq 100$ ). Следующие  $m$  строк содержат по три целых положительных числа каждая — города, которые соединяет соответствующая дорога и время, которое требуется для ее прохождения (время не превышает  $10^6$ ). По дороге можно перемещаться в любом из двух направлений, два города могут быть соединены несколькими дорогами.

#### Формат выходных данных

Если выполнить повеление короля невозможно, выведите на первой строке число  $-1$ . В противном случае выведите на первой строке минимальное возможное среднее время (с точностью 5 знаков после десятичной точки), которое требуется сыновьям, чтобы добраться до города назначения, не менее чем с пятью знаками после десятичной точки. В следующих  $k$  строках выведите пути сыновей, сначала число дорог в пути, и затем номера дорог в пути в том порядке, в котором их следует проходить. Дороги нумеруются, начиная с единицы, в том порядке, в котором они заданы во входном файле.

#### Пример

stdin	stdout
5 8 2	3.00000
1 2 1	3 1 5 6
1 3 1	3 2 7 8
1 4 3	
2 5 5	
2 3 1	
3 5 1	
3 4 1	
5 4 1	

#### Замечание

У вас есть решение A и декомпозиция потока со старого контеста...

### Задача 5С. $k$ паросочетаний [0.5 sec, 256 mb]

Дан полный взвешенный двудольный граф с равным количеством вершин в долях. Требуется выбрать  $k$  максимальных попарно не пересекающихся паросочетаний так, чтобы их суммарный вес был минимален.

#### Формат входных данных

Первая строка входного файла содержит  $n$  и  $k$  — количество вершин в каждой из долей и количество паросочетаний ( $2 \leq n \leq 50$ ,  $1 \leq k \leq n$ ). Каждая из последующих  $n$  строк содержит по  $n$  чисел:  $C_{ij}$  — вес ребра, ведущего из  $i$ -й вершины левой доли в  $j$ -ю правой.

Все значения во входном файле неотрицательны и не превосходят  $10^6$ .

#### Формат выходных данных

В первую строку выходного файла выведите одно число — искомый суммарный вес паросочетаний. Следующие  $k$  строк должны содержать  $n$  чисел — номера вершины, правой доли, соответствующие вершинам левой.

#### Примеры

stdin	stdout
3 2	6
1 2 1	1 2 3
1 1 2	3 1 2
2 1 1	

#### Замечание

Наконец-то делаем что-то полезное!

### Задача 5D. Автоматное программирование [0.3 sec, 256 mb]

В один замечательный день в компанию «X» 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

stdin	stdout
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

#### Замечание

В первом примере задания требуют выполнения в моменты времени  $2 \dots 8$ ,  $1 \dots 3$  и  $4 \dots 4$ , соответственно. Первое задание пересекается со вторым и третьим, поэтому можно выполнять либо его одно (прибыль 5), либо второе и третье (прибыль 6).

#### Подсказка по решению

Задача разобрана в практике. Чтобы пройти TL, нужно построить граф из  $\mathcal{O}(n)$  вершин и рёбер. Решение на таком графе с Дейкстрой точно сможет пройти TL (авторские 0.094 и 0.047 в зависимости от размера графа).

Можно строить граф из  $2n$  вершин, а можно из  $n$ .

84-й тест – специальный тест против Форда-Беллмана ; -)

## Дополнительные задачи

### Задача 5Е. Глобальный разрез [1.5 сек, 256 mb]

Дан взвешенный неориентированный граф. Нужно удалить минимальное по сумме весов множество ребер так, чтобы граф стал несвязным. После того, как граф становится несвязным, множество вершин распадается на два непустых множества  $S$  и  $T$ . Зная одно из двух множеств, можно восстановить второе и удаленные ребра.

#### Формат входных данных

Количество вершин и ребер в графе —  $2 \leq N \leq 500, 1 \leq M \leq 125\,000$ .

Далее  $M$  строк, в каждой 3 числа —  $1 \leq a, b \leq N$  и  $0 \leq w \leq 10^9$  (вершины, которые соединяет очередное ребро, и вес ребра).

#### Формат выходных данных

Минимальный суммарный вес ребер, которых нужно удалить.

Далее размер множества  $S$  и собственно вершины из  $S$ .

#### Пример

stdin	stdout
3 3 1 2 2 2 3 3 3 1 1	3 2 2 3
6 7 1 2 10 2 3 10 3 1 10 4 5 10 5 6 10 6 4 10 2 5 1	1 3 4 5 6

#### Замечание

Вы умеете Штор-Вагнера, умеете Каргера-Штейна. Выбирайте ;-)

### Задача 5F. Roads [1 sec, 256 mb]

The kingdom of Farland has  $N$  cities connected by  $M$  roads. Some roads are paved with stones, others are just country roads. Since paving the road is quite expensive, the roads to be paved were chosen in such a way that for any two cities there is exactly one way to get from one city to another passing only the stoned roads.

The kingdom has a very strong bureaucracy so each road has its own ordinal number ranging from 1 to  $M$ : the stoned roads have numbers from 1 to  $N - 1$  and other roads have numbers from  $N$  to  $M$ . Each road requires some money for support,  $i$ -th road requires  $c_i$  coins per year to keep it intact. Recently the king has decided to save some money and keep financing only some roads. Since he wants his people to be able to get from any city to any other, he decided to keep supporting some roads in such a way, that there is still a path between any two cities.

It might seem to you that keeping the stoned roads would be the good idea, however the king did not think so. Since he did not like to travel, he did not know the difference between traveling by a stoned road and travelling by a muddy road. Thus he ordered you to bring him the costs of maintaining the roads so that he could order his wizard to choose the roads to keep in such a way that the total cost of maintaining them would be minimal.

Being the minister of communications of Farland, you want to help your people to keep the stoned roads. To do this you want to fake the costs of maintaining the roads in your report to the king. That is, you want to provide for each road the fake cost of its maintaining  $d_i$  in such a way, that stoned roads form the set of roads the king would keep. However, to lower the chance of being caught, you want the sum  $\sum |c_i - d_i|$  to be as small as possible.

You know that the king's wizard is not a complete fool, so if there is the way to choose the minimal set of roads to be the set of the stoned roads, he would do it, so ties are allowed.

#### Формат входных данных

The first line of the input file contains  $N$  and  $M$  ( $2 \leq N \leq 60$ ,  $N - 1 \leq M \leq 400$ ). Next  $M$  lines contain three integer numbers  $a_i$ ,  $b_i$  and  $c_i$  each — the numbers of the cities the road connects ( $1 \leq a_i \leq N$ ,  $1 \leq b_i \leq N$ ,  $a_i \neq b_i$ ) and the cost of maintaining it ( $1 \leq c_i \leq 10\,000$ ).

#### Формат выходных данных

Output  $M$  lines — for each road output  $d_i$  that should be reported to be its maintainance cost so that the king would choose first  $N - 1$  roads to be the roads to keep and the sum  $\sum |c_i - d_i|$  is minimal possible.

#### Пример

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

#### Подсказка по решению

You have some hint in the theoretical part oh homework.