

## 1A. Паросочетание

2 секунды, 256 мегабайт

Двудольным графом называется неориентированный граф  $(V, E)$ ,  $E \subseteq V \times V$  такой, что его множество вершин  $V$  можно разбить на два множества  $A$  и  $B$ , для которых  $\forall (e_1, e_2) \in E \ e_1 \in A, e_2 \in B$  и  $A \cup B = V, A \cap B = \emptyset$ .

Паросочетанием в двудольном графе называется любой набор его несмежных рёбер, то есть такой набор  $S \subseteq E$ , что для любых двух рёбер  $e_1 = (u_1, v_1), e_2 = (u_2, v_2)$  из  $S$   $u_1 \neq u_2$  и  $v_1 \neq v_2$ .

Ваша задача — найти максимальное паросочетание в двудольном графе, то есть паросочетание с максимально возможным числом рёбер.

**Входные данные**

В первой строке записаны два целых числа  $n$  и  $m$  ( $1 \leq n, m \leq 250$ ), где  $n$  — число вершин в множестве  $A$ , а  $m$  — число вершин в  $B$ .

Далее следуют  $n$  строк с описаниями рёбер —  $i$ -я вершина из  $A$  описана в  $(i + 1)$ -й строке файла. Каждая из этих строк содержит номера вершин из  $B$ , соединённых с  $i$ -й вершиной  $A$ . Гарантируется, что в графе нет кратных ребер. Вершины в  $A$  и  $B$  нумеруются независимо (с единицы). Список завершается числом 0.

**Выходные данные**

Первая строка выходного файла должна содержать одно целое число  $l$  — количество рёбер в максимальном паросочетании. Далее следуют  $l$  строк, в каждой из которых должны быть два целых числа  $u_j$  и  $v_j$  — концы рёбер паросочетания в  $A$  и  $B$  соответственно.

входные данные
2 2 1 2 0 2 0
выходные данные
2 1 1 2 2

## 1B. День рождения

2 секунды, 256 мегабайт

Митя знаком с  $m$  юношами и  $n$  девушками и хочет пригласить часть из них на свой день рождения. Ему известно, с какими девушками знаком каждый юноша, и с какими юношами знакома каждая девушка. Он хочет добиться того, чтобы каждый приглашённый был знаком со всеми приглашёнными противоположного пола, пригласив при этом максимально возможное число своих знакомых. Помогите ему это сделать!

**Входные данные**

Входной файл состоит из одного или нескольких наборов входных данных. В первой строке входного файла записано число наборов  $k$  ( $1 \leq k \leq 20$ ). В последующих строках записаны сами наборы входных данных.

В первой строке каждого набора задаются числа  $0 \leq m \leq 150$  и  $0 \leq n \leq 150$ . Далее следуют  $m$  строк, в каждой из которых записано одно или несколько чисел — номера девушек, с которыми знаком  $i$ -й юноша (каждый номер встречается не более одного раза). Строка завершается числом 0.

**Выходные данные**

Для каждого набора выведите четыре строки. В первой из них выведите максимальное число знакомых, которых сможет пригласить Митя. В следующей строке выведите количество юношей и количество девушек в максимальном наборе знакомых, разделённые одним пробелом. Следующие две строки должны содержать номера приглашённых юношей и приглашённых девушек соответственно. Числа в каждой из этих двух строк разделяются ровно одним пробелом и выводятся в порядке возрастания. Если максимальных наборов несколько, то выведите любой из них.

Разделяйте вывод для разных наборов входных данных одной пустой строкой.

входные данные
2 2 2 1 2 0 1 2 0 3 2 1 2 0 2 0 1 2 0
выходные данные
4 2 2 1 2 1 2  4 2 2 1 3 1 2

## 1C. Толстые хоббиты

1 секунда, 64 мегабайта

Ни один хоббит не в состоянии в одиночку противостоять полчищам Мордора... В последний поход против Мордора Гэндальф решил отправить  $N$  хоббитов из Шира. Но часть хоббитов наотрез отказалась, жалуясь на то, что другие хоббиты наверняка будут дразнить их толстыми. После опроса всех хоббитов оказалось, что любой хоббит отказывается принять участие в походе в том случае, если с ним в поход выступит хотя бы один хоббит с меньшим весом. К счастью для Средиземья, не все хоббиты знают свой точный вес. В Шире были всего одни весы чашечного типа, позволяющие для пары хоббитов определить, какой хоббит тяжелее. Некоторые пары хоббитов взвешивались на этих весах. Всем хоббитам известен результат всех взвешиваний. Гэндальф абсолютно уверен, что в Шире нет двух хоббитов одного веса. Он заинтересован в том, чтобы отряд состоял из наибольшего количества хоббитов. Однако найти наибольшее множество хоббитов, среди которых ни один не считает себя тяжелее другого, оказалось не так-то просто. Подскажите Гэндальфу, на сколько хоббитов он может рассчитывать. Помните при этом, что хоббиты умные существа и знают, что если Сэм тяжелее Пиппина, а Пиппин тяжелее Фродо, то Сэм и подавно будет тяжелее Фродо.

**Входные данные**

В первой строке дано целое число  $N$  — количество хоббитов ( $2 \leq N \leq 100$ ). Все хоббиты пронумерованы целыми числами от 1 до  $N$ . В следующих  $N$  строках записана матрица размера  $N \times N$ . Если  $i$ -й и  $j$ -й хоббит взвешивались на чашечных весах и оказалось, что  $i$ -й хоббит тяжелее, то в  $i$ -й строке матрицы на  $j$ -й позиции стоит единица. Во всех остальных случаях в матрице стоят нули.

**Выходные данные**

В первой строке выведите размер наибольшего множества хоббитов, готового выступить в поход, во второй строке перечислите номера хоббитов из этого множества через пробел.

<b>входные данные</b>
2 0 1 0 0
<b>выходные данные</b>
1 2

<b>входные данные</b>
3 0 0 0 0 0 0 0 0 0
<b>выходные данные</b>
3 1 2 3

## 1D. Сокровищница

4 секунды, 512 мегабайт

Иэн и Барли добрались до цели своего путешествия. Им осталось лишь открыть сокровищницу, в которой их ждёт ещё один волшебный кристалл.

Латинским квадратом называется квадратная таблица размера  $x \times x$ , в которой ровно  $x$  различных значений, и в каждой строке и каждом столбце все значения различны.

Из древнего манускрипта, братьям известна последовательность целых чисел  $a_1 < a_2 < \dots < a_{n-1} < a_n$ . Чтобы скровищница открылась, нужно нарисовать на входе в неё квадратную таблицу размера  $a_n \times a_n$ , заполненную числами от 1 до  $a_n$ . При этом, для всех  $i$  подтаблица размера  $a_i \times a_i$ , верхний левый угол которой совпадает с верхним левым углом всей таблицы, должна являться латинским квадратом.

Помогите братьям нарисовать правильную таблицу, либо сообщите, что это невозможно.

### Входные данные

В первой строке дано одно целое число  $n$  — длина последовательности чисел ( $1 \leq n \leq 1\,000$ ).

Во второй строке даны  $n$  целых чисел  $a_i$  ( $1 \leq a_i \leq 1\,000$ ,  $a_i < a_{i+1}$ ).

### Выходные данные

Если искомая таблица существует, в первой строке выведите «Yes», а в следующих  $a_n$  строках по  $a_n$  чисел со значениями от 1 до  $a_n$  — таблицу. Если решений несколько, выведите любое.

Если искомой таблицы не существует, в единственной строке выведите «No».

### Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Ограничения	Необходимые подзадачи	Информация о проверке
1	10	$n = 1$		первая ошибка
2	15	$n \leq 10$ , $a_i = 2^{i-1}$		первая ошибка
3	15	$a_i \cdot 2 + 1 \geq a_{i+1}$ для всех $1 \leq i < n$	1, 2	первая ошибка
4	35	$a_n \leq 100$		первая ошибка
5	25	Без дополнительных ограничений	1, 2, 3, 4	первая ошибка

<b>входные данные</b>
1 3
<b>выходные данные</b>
Yes 1 3 2 2 1 3 3 2 1

<b>входные данные</b>
3 1 2 4
<b>выходные данные</b>
Yes 1 2 3 4 2 1 4 3 3 4 1 2 4 3 2 1

<b>входные данные</b>
2 2 3
<b>выходные данные</b>
No

<b>входные данные</b>
2 2 5
<b>выходные данные</b>
Yes 1 2 3 5 4 2 1 4 3 5 5 4 1 2 3 4 3 5 1 2 3 5 2 4 1

## 2A. Просто поток

5 секунд, 1024 мегабайта

Дана система из узлов и труб, по которым может течь вода. Для каждой трубы известна наибольшая скорость, с которой вода может протекать через нее. Известно, что вода течет по трубам таким образом, что за единицу времени в каждый узел (за исключением двух — источника и стока) втекает ровно столько воды, сколько из него вытекает.

Ваша задача — найти наибольшее количество воды, которое за единицу времени может протекать между источником и стоком, а также скорость течения воды по каждой из труб.

Трубы являются двусторонними, то есть вода в них может течь в любом направлении. Между любой парой узлов может быть более одной трубы.

### Входные данные

В первой строке записано натуральное число  $N$  — количество узлов в системе ( $2 \leq N \leq 100$ ). Известно, что источник имеет номер 1, а сток номер  $N$ . Во второй строке записано натуральное  $M$  ( $1 \leq M \leq 5000$ ) — количество труб в системе. Далее в  $M$  строках идет описание труб. Каждая труба задается тройкой целых чисел  $A_i, B_i, C_i$ , где  $A_i, B_i$  — номера узлов, которые соединяет данная труба ( $A_i \neq B_i$ ), а  $C_i$  ( $0 \leq C_i \leq 10^4$ ) — наибольшая допустимая скорость течения воды через данную трубу.

### Выходные данные

В первой строке выведите наибольшее количество воды, которое протекает между источником и стоком за единицу времени. Далее выведите  $M$  строк, в каждой из которых выведите скорость течения воды по соответствующей трубе. Если направление не совпадает с порядком узлов, заданным во входных данных, то выводите скорость со знаком минус. Числа выводите с точностью  $10^{-3}$ .

входные данные
2 2 2 1 2 1 2 1 3
выходные данные
4 1 -3

## 2B. Разрез

2 секунды, 1024 мегабайта

Найдите минимальный разрез между вершинами 1 и  $n$  в заданном неориентированном графе.

### Входные данные

На первой строке входного файла содержится  $n$  ( $2 \leq n \leq 100$ ) — число вершин в графе и  $m$  ( $0 \leq m \leq 400$ ) — количество ребер. На следующих  $m$  строках входного файла содержится описание ребер. Ребро описывается номерами вершин, которые оно соединяет, и его пропускной способностью (положительное целое число, не превосходящее 10 000 000), при этом никакие две вершины не соединяются более чем одним ребром.

### Выходные данные

На первой строке выходного файла должны содержаться количество ребер в минимальном разрезе и их суммарная пропускная способность. На следующей строке выведите возрастающую последовательность номеров ребер (ребра нумеруются в том порядке, в каком они были заданы во входном файле).

входные данные
3 3 1 2 3 1 3 5 3 2 7
выходные данные
2 8 1 2

## 2C. Великая стена

2 секунды, 1024 мегабайта

У короля Людовика двое сыновей. Они ненавидят друг друга, и король боится, что после его смерти страна будет уничтожена страшными войнами. Поэтому Людовик решил разделить свою страну на две части, в каждой из которых будет властвовать один из его сыновей. Он посадил их на трон в города  $A$  и  $B$ , и хочет построить минимально возможное количество фрагментов стены таким образом, чтобы не существовало пути из города  $A$  в город  $B$ .

Страну, в которой властвует Людовик, можно упрощенно представить в виде прямоугольника  $m \times n$ . В некоторых клетках этого прямоугольника расположены горы, по остальным же можно свободно перемещаться. Кроме этого, ландшафт в некоторых клетках удобен для строительства стены, в остальных же строительство невозможно.

При поездках по стране можно перемещаться из клетки в соседнюю по стороне, только если ни одна из этих клеток не содержит горы или построенного фрагмента стены.

### Входные данные

В первой строке входного файла содержатся числа  $m$  и  $n$  ( $1 \leq m, n \leq 50$ ). Следующие  $m$  строк по  $n$  символов задают карту страны. Символы обозначают: «#» — гора, «. » — место, пригодное для постройки стены, «-» — место, не пригодное для постройки стены, «A» и «B» — города  $A$  и  $B$ .

### Выходные данные

В первой строке выходного файла должно быть выведено минимальное количество фрагментов стены  $F$ , которые необходимо построить. Далее нужно вывести карту в том же формате, как во входном файле. Клетки со стеной обозначьте символом «+».

Если невозможно произвести требуемую застройку, то выведите в выходной файл единственное число - 1.

входные данные
5 5 ---.. A-.#- .#.#- ---.. --.-B
выходные данные
3 --+. . A-+#- +#.#- ---.. --.-B

входные данные
1 2 AB
выходные данные
-1

входные данные
2 2 A# #B
выходные данные
0 A# #B

## 2D. Плотник

2 секунды, 256 мегабайт

Такеши — известный плотник, он принимает много заказов со всей Японии. Однако недавно воры украли у него все инструменты, и теперь ему придется купить новые, чтобы выполнять заказы.

В магазине инструментов Такеши узнал цены на каждый из  $m$  интересующих его инструментов, кроме того, он выяснил, что на некоторые пары инструментов есть скидка, если их покупать вместе. При этом каждый инструмент встречается не более чем в одной такой паре.

Сейчас у Такеши есть  $n$  заказов, для каждого заказа он знает его стоимость и список инструментов, которые нужны, чтобы его выполнить.

Такеши необязательно выполнять все заказы. Он хочет выбрать какое-то множество заказов и купить только те инструменты, которые нужны для их выполнения. При этом он хочет максимизировать свою прибыль, то есть разницу между стоимостью выполненных заказов и стоимостью инструментов с учетом скидок. Помогите Такеши сделать это.

### Входные данные

Первая строка ввода содержит числа  $n$ ,  $m$  и  $p$  — число заказов, число инструментов и число скидок ( $1 \leq n \leq 100$ ,  $2 \leq m \leq 100$ ,  $1 \leq p \leq m / 2$ ).

Следующие  $n$  строк содержат описание заказов. Каждое описание начинается со стоимости заказа (положительное целое число не больше 1000), далее идет число инструментов, необходимых для его выполнения и номера этих инструментов.

Следующие  $m$  строк содержат стоимости инструментов без скидки (положительные целые числа не больше 1000).

Следующие  $p$  строк содержат описание скидок. Каждое описание содержит номера инструментов и их суммарную стоимость с учетом скидки. Гарантируется, что каждый инструмент встречается только в одной скидке. Также гарантируется, что сумма с учетом скидки больше стоимости каждого из инструментов по отдельности и меньше их суммы.

**Выходные данные**  
Выведите максимальную прибыль, которую может получить Такеши.

входные данные
3 4 2 100 2 1 2 100 1 3 100 1 4 20 20 50 150 1 2 30 3 4 180
выходные данные
120

Условие  
недоступно  
на  
русском  
языке

2F. Пение

2 секунды, 256 мегабайт

Алиса и Боб собираются спеть песню вместе. Вам даны три целых числа:  $n$ ,  $low$  и  $high$ . Для простоты, обозначим числами от 1 до  $n$  высоты нот, которые встречаются в песне (от самых низких до самых высоких). Алиса может петь только ноты с высотами от  $low$  до  $n$  включительно. Боб может петь только ноты с высотами от 1 до  $high$  включительно. (Гарантируется, что эти отрезки пересекаются).

Вам дана последовательность высот нот в песне. Каждая нота в песне должна быть спета ровно одним человеком: или Алисой, или Бобом. Все ноты одной высоты должны быть спеты одним человеком. Количество раз, которое сменяется певец, в течении песни должно быть минимальным возможным.

Формально, смена певца происходит в тот момент, когда одну ноту пел один человек, а следующую — уже другой.

Вычислите минимально возможное количество смен певца в течении песни.

**Входные данные**

В первой строке даны четыре целых числа  $n$ ,  $low$ ,  $high$  и  $m$  ( $1 \leq low \leq high \leq n \leq 1000$ ,  $1 \leq m \leq 1000$ ).

Во второй строке даны  $m$  чисел — последовательность нот из песни  $h_i$  ( $1 \leq h_i \leq n$ ).

**Выходные данные**

Выведите единственное число — ответ на задачу.

входные данные
3 2 2 6 1 2 3 2 1 2
выходные данные
2

входные данные
10 3 7 8 4 4 5 5 6 5 3 6
выходные данные
0

входные данные
6 2 5 6 5 3 1 6 4 2
выходные данные
1

входные данные
10 4 5 9 1 4 3 5 2 5 7 5 9
выходные данные
3

входные данные
100 20 80 24 2 27 3 53 53 52 52 60 85 89 100 53 60 2 3 53 100 89 40 42 2 53 2 85
выходные данные
5

2G. Два языка

2 секунды, 256 мегабайт

Родители Эллиота говорят с ним дома по-английски и по-французски. Он слышал множество разных слов, но не всегда уверен, какие слова принадлежат какому языку. Эллиот знает одно предложение, про которое точно уверен, что оно на английском, и знает одно предложение, про которое уверен, что оно на французском. И еще несколько предложений, которые могут быть или на английском, или на французском.

Если слово встречается в английском предложении, оно должно быть словом в английском языке. Если слово встречается во французском предложении, оно должно быть словом во французском языке.

Вам даны все предложения, которые слышал Эллиот. Найдите минимальное возможное количество слов, среди тех, которые он слышал, которые должны принадлежать сразу обоим языкам.

**Входные данные**

В первой строке содержится количество тестов  $T$  ( $1 \leq T \leq 25$ ). Далее следует  $T$  тестов. Каждый начинается со строки, содержащей целое число  $N$  ( $1 \leq N \leq 200$ ). Далее следует  $N$  строк, каждая содержит последовательность слов, разделенных пробелом. Каждое слово состоит только из строчных латинских букв. Первая из этих  $N$  строк — предложение на английском. Вторая — предложение на французском. Остальные могут быть как предложениями на английском, так и предложениями на французском.

Каждое слово содержит не более 10 символов. Первые два предложения содержат не более 1 000 слов каждое. Остальные содержат не более чем по 10 слов.

**Выходные данные**

Для каждого теста выведите в новой строке одно число — ответ на задачу.

входные данные
4 2 he loves to eat baguettes il aime manger des baguettes 4 a b c d e f g h i j a b c i j f g h d e 4 he drove into a cul de sac elle a conduit sa voiture il a conduit dans un cul de sac il mange pendant que il conduit sa voiture 6 adieu joie de vivre je ne regrette rien adieu joie de vivre je ne regrette rien a b c d e f g h i j a b c i j f g h d e
выходные данные
1 4 3 8

3А. Мороженое

2 секунды, 256 мегабайт

По дороге в школу Петя любит забегать в киоск и покупать себе мороженое. Однако при этом он часто опаздывает в школу. Неожиданно Петя понял — он просто ходит не по кратчайшему пути!

Помогите Пете победить опоздания. Город можно представить как  $n$  перекрестков, соединенных  $m$  улицами, про каждую улицу известна ее длина. Дом Пети находится на перекрестке  $a$ , школа — на перекрестке  $b$ , а киоск с мороженым — на перекрестке  $c$ . По пути в школу Петя никогда не проходит через один перекресток дважды.

Входные данные

Первая строка входного файла содержит числа  $n$  и  $m$  ( $3 \leq n \leq 3 \cdot 10^4, 0 \leq m \leq 5 \cdot 10^4$ ). Вторая строка содержит три различных числа —  $a, b$  и  $c$ . Следующие  $m$  строк содержат по три целых числа  $x_i, y_i$  и  $l_i$  — номера перекрестков, соединенных улицей и ее длину (длина — целое неотрицательное число, которое не превышает  $10^4$ ).

Выходные данные

Если путь из дома Пети до школы, проходящие через перекресток с мороженым и не проходящий по одному перекрестку два раза, существует, выведи на первой строке выходного файла два целых числа  $k$  и  $l$  — количество улиц, которые проходит Петя в оптимальном пути и длину пути. На второй строке выведите номера перекрестков в том порядке, в котором их посещает Петя.

В противном случае выведите - 1 на первой строке выходного файла.

входные данные
5 6 1 5 3 1 2 1 2 3 4 2 4 1 3 4 2 1 5 1 4 5 2
выходные данные
4 9 1 2 3 4 5

3В. Задача о назначениях

2 секунды, 256 мегабайт

{256 мегабайт}

Дана целочисленная матрица  $C$  размера  $n \times n$ . Требуется выбрать  $n$  ячеек так, чтобы в каждой строке и каждом столбце была выбрана ровно одна ячейка, а сумма значений в выбранных ячейках была минимальна.

Входные данные

Первая строка входного файла содержит  $n$  ( $2 \leq n \leq 300$ ). Каждая из последующих  $n$  строк содержит по  $n$  чисел:  $C_{ij}$ . Все значения во входном файле неотрицательны и не превосходят  $10^6$ .

Выходные данные

В первую строку выходного файла выведите одно число — искомая минимизируемая величина. Далее выведите  $n$  строк по два числа в каждой — номер строки и столбца клетки, участвующей в оптимальном назначении.

Пары чисел можно выводить в произвольном порядке.

входные данные
3 3 2 1 1 3 2 2 1 3

выходные данные

3 1 3 2 1 3 2
------------------------

3С. План эвакуации

2 секунды, 256 мегабайт

В городе есть муниципальные здания и бомбоубежища, которые были специально построены для эвакуации служащих в случае ядерной войны. Каждое бомбоубежище имеет ограниченную вместительность по количеству людей, которые могут в нем находиться. В идеале все работники из одного муниципального здания должны были бы бежать к ближайшему бомбоубежищу. Однако, в таком случае, некоторые бомбоубежища могли бы переполниться, в то время как остальные остались бы наполовину пустыми.

Чтобы разрешить эту проблему Городской Совет разработал специальный план эвакуации. Вместо того, чтобы каждому служащему индивидуально приписать, в какое бомбоубежище он должен бежать, для каждого муниципального здания определили, сколько служащих из него в какое бомбоубежище должны бежать. Задача индивидуального распределения была переложена на внутреннее управление муниципальных зданий.

План эвакуации учитывает количество служащих в каждом здании — каждый служащий должен быть учтен в плане и в каждое бомбоубежище может быть направлено количество служащих, не превосходящее вместимости бомбоубежища.

Городской Совет заявляет, что их план эвакуации оптимален в том смысле, что суммарное время эвакуации всех служащих города минимально.

Мэр города, находящийся в постоянной конфронтации с Городским Советом, не слишком то верит этому заявлению. Поэтому он нанял Вас в качестве независимого эксперта для проверки плана эвакуации. Ваша задача состоит в том, чтобы либо убедиться в оптимальности плана Городского Совета, либо доказать обратное, представив в качестве доказательства другой план эвакуации с меньшим суммарным временем для эвакуации всех служащих.

Карта города может быть представлена в виде квадратной сетки. Расположение муниципальных зданий и бомбоубежищ задается парой целых чисел, а время эвакуации из муниципального здания с координатами  $(X_i, Y_i)$  в бомбоубежище с координатами  $(P_j, Q_j)$  составляет  $D_{ij} = |X_i - P_j| + |Y_i - Q_j| + 1$  минут.

Входные данные

Входной файл содержит описание карты города и плана эвакуации, предложенного Городским Советом. Первая строка входного файла содержит два целых числа  $N$  ( $1 \leq N \leq 100$ ) и  $M$  ( $1 \leq M \leq 100$ ), разделенных пробелом.  $N$  — число муниципальных зданий в городе (все они занумерованы числами от 1 до  $N$ ),  $M$  — число бомбоубежищ (все они занумерованы числами от 1 до  $M$ ).

Последующие  $N$  строк содержат описания муниципальных зданий. Каждая строка содержит целые числа  $X_i, Y_i$  и  $B_i$ , разделенные пробелами, где  $X_i, Y_i$  ( $-1000 \leq X_i, Y_i \leq 1000$ ) — координаты здания, а  $B_i$  ( $1 \leq B_i \leq 1000$ ) — число служащих в здании.

Описание бомбоубежищ содержится в последующих  $M$  строках. Каждая строка содержит целые числа  $P_j, Q_j$  и  $C_j$ , разделенные пробелами, где  $P_j, Q_j$  ( $-1000 \leq P_j, Q_j \leq 1000$ ) — координаты бомбоубежища, а  $C_j$  ( $1 \leq C_j \leq 1000$ ) — вместимость бомбоубежища.

В последующих  $N$  строках содержится описание плана эвакуации. Каждая строка представляет собой описание плана эвакуации для отдельного здания. План эвакуации из  $i$ -го здания состоит из  $M$  целых чисел  $E_{ij}$ , разделенных пробелами.  $E_{ij}$  ( $0 \leq E_{ij} \leq 10000$ ) — количество служащих, которые должны эвакуироваться из  $i$ -го здания в  $j$ -е бомбоубежище.

Гарантируется, что план, заданный во входном файле, корректен.

Выходные данные



Если план эвакуации Городского Совета оптимален, то выведите одно слово `OPTIMAL`. В противном случае выведите на первой строке слово `SUBOPTIMAL`, а в последующих  $N$  строках выведите Ваш план эвакуации (более оптимальный) в том же формате, что и во входном файле. Ваш план не обязан быть оптимальным, но должен быть лучше плана Городского Совета.

входные данные
3 4 -3 3 5 -2 -2 6 2 2 5 -1 1 3 1 1 4 -2 -2 7 0 -1 3 3 1 1 0 0 0 6 0 0 3 0 2
выходные данные
SUBOPTIMAL 3 0 1 1 0 0 6 0 0 4 0 1

входные данные
3 4 -3 3 5 -2 -2 6 2 2 5 -1 1 3 1 1 4 -2 -2 7 0 -1 3 3 0 1 1 0 0 6 0 0 4 0 1
выходные данные
OPTIMAL

3D. Задача коммивояжеров

2 секунды, 256 мегабайт

Есть  $n$  городов. Между городами есть ориентированные дороги, у каждой дороги есть стоимость покупки разрешения на проезд. Мы хотим торговать во всех городах. У нас есть неограниченное кол-во коммивояжеров. Для каждого из них мы должны определить список городов, в которых они будут торговать. Каждый коммивояжер будет объезжать все города из своего списка по циклу (он может по пути заезжать в другие города, но не торговать там). Если два (или более) коммивояжеров будут ездить по одной дороге, то каждому из них мы должны купить разрешение на проезд. Если список у коммивояжера состоит только из одного города, то он либо должен регулярно выезжать из города (тоже по какому-то циклу), либо мы должны купить ему прописку (у каждого города есть цена прописки). Наконец, в любом городе должен торговать только один коммивояжер, иначе предприятием заинтересуется налоговая. Нужно минимизировать издержки.

Входные данные

В первой строке два числа  $n, m$  — количество городов и количество дорог ( $1 \leq n \leq 256, 0 \leq m \leq n(n - 1)$ ).

Во второй строке  $n$  чисел  $a_i$  — цена прописки для города номер  $i$  ( $0 \leq a_i \leq 10^9$ ).

Затем в  $m$  строках описаны дороги. Описание дороги из города  $u$  в город  $v$  со стоимостью разрешения на проезд  $c$  выглядит как  $u\ v\ cost$  ( $1 \leq u, v \leq n, u \neq v, 0 \leq c \leq 10^9$ ). Гарантируется, что между любой парой городов не более 1 дороги в каждом из направлений.

Выходные данные

Выведите одно число — минимальную сумму издержек.

входные данные
3 3 30 25 30 1 2 3 2 3 5 3 1 10
выходные данные
18

4A. Very simple problem

10 секунд, 256 мегабайт

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

$$c_1x_1 + \dots + c_nx_n \rightarrow \max$$

Входные данные

В первой строк входных данных содержится два целых числа:  $n$  и  $m$  — количество переменных и количество уравнений, соответственно ( $1 \leq n \leq 600, 1 \leq m \leq 1000$ ). Следующие  $m$  строк содержат описания уравнений, описание  $i$ -ого уравнения содержит вещественные числа  $a_{i1}, \dots, a_{in}$  и  $b_i$ . Последняя строка содержит  $n$  вещественных чисел:  $c_1, \dots, c_n$ . Все  $a_{ij}$  не превосходят  $10^3$  по модулю. Все  $b_i$  и  $c_i$  не превосходят  $10^9$  по модулю.

Выходные данные

В случае, если данная задача не имеет допустимых решений, выведите `No solution`. Если же можно получить неограниченно большое значение целевой функции, выведите `Unbounded`.

В противном случае в первой строке выведите слово `Bounded`. Во второй строке выведите значения переменных, для которых достигается максимум целевой функции. Ваш ответ будет считаться правильным, если значение целевой функции в вашей точке будет отличаться от правильного не более, чем на  $10^{-6}$  по абсолютной или относительной погрешности.

5A. Задача для второклассника

2 секунды, 256 мегабайт

Вам даны два числа. Необходимо найти их произведение.

Входные данные

Входные данные состоят из двух строк, на каждой из которых находится целое одно **целое** число, длина которого не превосходит двухсот пятидесяти тысяч символов.

Выходные данные

Выведите произведение данных чисел.

входные данные
2 2
выходные данные
4

входные данные
1 -1
выходные данные
-1

