

Задача A. Anniversary

Имя входного файла: **a.in**
Имя выходного файла: **a.out**
Ограничение по времени: **1 second**
Ограничение по памяти: **64 Mebibytes**

В 2134 году, когда население Земли достигло 10^{18} человек, было решено отметить это событие следующим образом. На огромной территории, специально отведенной под празднование, было размещено N миниатюрных лампочек (линейные размеры одной лампочки составляли менее 0.1 миллиметра), некоторым образом пронумерованных последовательными числами от 1 до N .

Изначально все лампочки были выключены. Далее было произведено ровно 10^{18} шагов — по одному в честь каждого жителя Земли. На i -м шаге одновременно изменялось состояние всех лампочек, номера которых делятся нацело на i . Изменение состояния означает, что если лампочка была выключена, то она становится включенной, и наоборот, если лампочка была включена, то она становится выключенной. Промежуток времени между последовательными шагами составил 1 пикосекунду, таким образом, все празднование заняло порядка полутора недель.

Вообще, для сторонних наблюдателей все это выглядело как беспорядочное бессмысленное мерцание, тем не менее, все были в восторге — настолько очевиден был грандиозный масштаб этого замечательного мероприятия!

Но вот все закончилось. Смотреть стало не на что и, как показал более трезвый анализ, смотреть, собственно, и было не на что.

Тем временем, после выполнения шага 10^{18} , некоторые лампочки остались во включенном состоянии. Пока жители Земли отходят от шока, размышляя о том, зачем им понадобилось подобное празднование и как теперь покрыть баснословные затраты на всю эту впечатляющую ерунду, вам предлагается посчитать количество лампочек, которые все еще горят и расходуют драгоценную электроэнергию.

Формат входного файла

Первая строка входного файла содержит целое число N ($1 \leq N \leq 2^{63} - 1$).

Формат выходного файла

Выведите единственное целое число — количество лампочек, которые остались во включенном состоянии по завершении шага с номером 10^{18} .

Пример

a.in	a.out
1	1
9	3
100	10

Задача В. Trade

Имя входного файла: **b.in**
Имя выходного файла: **b.out**
Ограничение по времени: **1 second**
Ограничение по памяти: **64 Mebibytes**

Торговля — штука тонкая. Успешный торгош должен уметь не только вовремя продавать нужные товары и заправски зазывать клиента, но и досконально знать ситуацию на рынке. Среди всего прочего важно знать, кто из других торговцев торгует друг с другом, а кто нет. Бывает, что торговцы не общаются напрямую, но их товар всё равно попадает друг к другу через других торговцев. Например, если торговцы А и В торгуются напрямую, и торговцы В и С торгуются напрямую, то товары А и С будут попадать друг к другу через торговца В. В общем случае товар может попасть от одного торговца к другому не напрямую через любое количество торговцев. Очень важно понятие *неразлучных пар* — это такие пары торговцев, которые торгуют напрямую, и нет других торговцев, через которых эта пара смогла бы обмениваться товаром не напрямую.

Манао хочет стать успешным торговцем. Мы не знаем, какими необходимыми качествами он обладает, но знания о ситуации на рынке ему определённо не хватает. Всё, что ему известно — что на рынке есть N торговцев, M их пар торгуют напрямую, а K из этих пар являются к тому же неразлучными. А требуется ему более конкретная информация в стиле “А торгует с В, С торгует с D, X торгует с А”. Это всё, конечно, не всегда однозначно, но на данный момент любой подходящий под описание расклад сойдёт.

Вам дают T сценариев, в каждом из которых свои значения N , M и K . Для каждого определите, соответствует ли он какой-либо торговой сети и если да, то выведите её описание. Торговцев пронумеруем в каком-либо порядке от 1 до N , а описанием торговой сети назовём все различные пары торговцев, торгующих напрямую.

Формат входного файла

Первая строка содержит количество сценариев T . Далее следует T строк, каждая из которых содержит три числа N , M , K ($2 \leq N \leq 100$, $0 \leq K \leq M \leq N \cdot (N - 1)/2$).

Количество сценариев в одном входном файле не превышает 100. Сумма M по всем сценариям в одном входном файле не превышает $5 \cdot 10^4$.

Формат выходного файла

Для каждого из T сценариев выведите “NO SOLUTION” (без кавычек), если соответствующей торговой сети не существует. В противном случае выведите “TRADE MARKET FOUND”, а далее M строк. Каждая из строк должна содержать пару чисел, разделённых пробелом — номера торговцев, обменивающихся товаром напрямую.

Пример

b.in	b.out
3	TRADE MARKET FOUND
4 3 0	1 2
4 2 0	2 3
5 5 2	3 1
	NO SOLUTION
	TRADE MARKET FOUND
	1 2
	2 3
	3 1
	1 4
	1 5

Задача C. Grid

Имя входного файла: `c.in`
Имя выходного файла: `c.out`
Ограничение по времени: 2 seconds
Ограничение по памяти: 64 Mebibytes

На плоскости задана бесконечная сетка из вертикальных и горизонтальных прямых. Эти прямые заданы уравнениями $X = i \cdot K$ и $Y = j \cdot K$ для любых целых чисел i и j . Также задано N точек. Для конкретной точки мы говорим, что она *принадлежит сетке*, если эта точка принадлежит хотя бы одной из прямых, образующих сетку.

Есть возможность двигать сетку параллельно осям координат. Перенос сетки на вектор (dx, dy) означает, что вместо каждой из прямых $X = i \cdot K$ возникает прямая $X = i \cdot K + dx$, а вместо каждой из прямых $Y = j \cdot K$ возникает прямая $Y = j \cdot K + dy$. Найдите вектор переноса сетки, в результате применения которого ей будет принадлежать наибольшее количество заданных точек.

Формат входного файла

Первая строка содержит два целых числа N и K ($1 \leq N \leq 10^5$, $2 \leq K \leq 10^9$). i -ая из следующих N строк содержит пару чисел X_i, Y_i — координаты i -ой точки. Координаты всех точек целые и по абсолютной величине не превосходят 10^9 . Никакие две точки не совпадают.

Формат выходного файла

Выведите максимальное возможное количество одновременно принадлежащих сетке точек из заданного множества.

Пример

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

Note

Один из возможных векторов переноса сетки в данном примере — $(1, 0)$.

Задача D. Drawing

Имя входного файла: `d.in`
Имя выходного файла: `d.out`
Ограничение по времени: 1 second
Ограничение по памяти: 64 Mebibytes

Маленький Вася очень любит рисовать.

Сегодня он взял таблицу из N строк и M столбцов и решил раскрасить ее клетки в разные цвета. Чтобы результат раскраски получился менее предсказуемым, он решил произвести ее следующим образом:

- Для каждого i от 1 до N с вероятностью $R_i\%$ Вася полностью раскрашивает все клетки i -й строки в красный цвет.
- Для каждого j от 1 до M с вероятностью $C_j\%$ Вася полностью раскрашивает все клетки j -го столбца в синий цвет.

Если клетка раскрашивается и в красный, и в синий цвета, то в результате смешения цветов она оказывается зеленой.

Изначально ни одна из клеток таблицы раскрашена не была, а по окончании применения процедуры, описанной выше, G клеток оказались закрасенными в зеленый цвет. Определите математическое ожидание общего количества клеток, раскрашенных Васей в какой-либо цвет (красный, синий или зеленый).

Формат входного файла

Первая строка входного файла содержит три целых числа N , M и G , разделенные одиночными пробелами ($1 \leq N$, $M \leq 100$, $0 \leq G \leq N \cdot M$).

Во второй строке записаны целые числа R_1, R_2, \dots, R_N , разделенные одиночными пробелами. В третьей строке записаны целые числа C_1, C_2, \dots, C_M , разделенные одиночными пробелами ($0 < R_i, C_i < 100$).

Гарантируется, что существует хотя бы один способ раскрасить некоторые строки в красный цвет, а некоторые столбцы — в синий так, что ровно G клеток окажутся закрасенными в зеленый цвет.

Формат выходного файла

Выведите искомое математическое ожидание. Абсолютная погрешность выведенного ответа не должна превышать 10^{-5} .

Пример

d.in	d.out
2 2 1 50 50 50 50	3.0
2 2 0 50 50 50 50	2.2857142857142856
2 3 2 3 2 3 2 3	5.313107297058556

Задача E. Empire

Имя входного файла: e.in
Имя выходного файла: e.out
Ограничение по времени: 3 seconds
Ограничение по памяти: 64 Mebibytes

Империя включает в себя N планет. Пронумеруем эти планеты числами от 1 до N . Планета с номером 1 — столица Империи, где находится резиденция Императора и подготавливаются войска. На различных планетах империи часто вспыхивают восстания, которые приходится подавлять военной силой и немедленно.

Для того, чтобы войска могли быстро передвигаться, на некоторых планетах установлены односторонние телепорты. Всего телепортов M штук. С помощью i -ого телепорта можно в мгновение ока попасть с планеты A_i на планету B_i (но не наоборот). Таким образом, вовремя подавить восстание на некоторой планете X можно, если существует последовательность планет P_1, \dots, P_k ($k \geq 2$) такая, что $P_1 = 1$, $P_k = X$, а для каждого $1 \leq i < k$ есть телепорт из планеты с номером P_i на планету с номером P_{i+1} . Учитывая, что после подавления восстания войска остаются на планете для поддержания порядка, об их возвращении в столицу мы можем не беспокоиться.

Проверьте, есть ли возможность при имеющихся телепортах подавить восстание на любой планете Империи. Если это так, выведите 0. В противном случае, найдите наименьшее количество телепортов, которое надо дополнительно построить, чтобы восстание на любой планете было подавляемо. Каждый телепорт может быть построен между произвольными двумя планетами.

Формат входного файла

Первая строка содержит два целых числа N и M ($2 \leq N \leq 10^5$, $0 \leq M \leq 2 \cdot 10^5$).

i -ая из следующих M строк содержит пару чисел A_i, B_i ($1 \leq A_i, B_i \leq N$ для всех $1 \leq i \leq M$).

Ни на одной планете нет телепорта в саму себя.

Ни на одной планете нет двух телепортов на одну и ту же планету.

Формат выходного файла

Выведите наименьшее количество дополнительных телепортов, гарантирующее, что восстание на любой планете можно будет немедленно подавить.

Пример

e.in	e.out
6 4	2
3 1	
4 6	
1 2	
4 5	

Note

Можно построить телепорт из планеты 2 на планету 4 и из планеты 5 на планету 3.

Задача F. Reform (Division 1 Only!)

Имя входного файла: `f.in`
Имя выходного файла: `f.out`
Ограничение по времени: 3 seconds
Ограничение по памяти: 64 Mebibytes

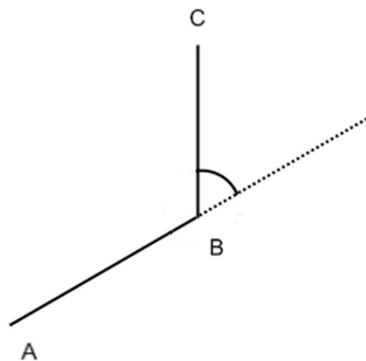
На одной замечательной плоскости с декартовой системой координат испокон веков была расположена очень консервативная страна. В этой стране испокон веков было N городов, пронумерованных последовательными целыми числами от 1 до N . По сравнению с бесконечностью плоскости города, даже очень большие, очень невелики, поэтому мы будем считать их точками на плоскости. Координаты i -го города — (X_i, Y_i) . Координаты любых двух городов различны. Никакие 3 города не лежат на одной прямой линии.

Некоторые пары городов соединены двунаправленными дорогами. Каждая дорога — это отрезок прямой линии, соединяющий некоторые два города. Известно, что из каждого города выходит ровно 3 дороги. Никакая дорога не соединяет город с самим собой. Между каждой парой городов может быть не более одной дороги.

Так обстояли дела в этой стране с незапамятных времен, и никому даже не приходило в голову что-либо поменять. Но вот произошла беда — к власти пришел либеральный король! И проблемы не заставили себя ждать — незамедлительно последовал указ о реформе дорожного сообщения в стране. Было приказано убрать некоторые дороги так, чтобы в результате:

- Из каждого города выходило ровно 2 дороги.
- Угол поворота между двумя дорогами, выходящими из одного и того же города, был *строго* меньше 60 градусов.
- Никакие две дороги не пересекались нигде, кроме как в городах.

Угол поворота между двумя дорогами вычисляется следующим образом. Пусть из города B дороги идут в города A и C . Тогда угол поворота между ними равен внешнему углу при вершине B в треугольнике ABC (см. рисунок).



Реализация реформы была поручена министру транспорта. Именно ему предстоит решить, какие дороги убрать, а какие оставить. Желая угодить королю, среди всех возможных способов решения поставленной королем задачи, министр хочет выбрать такой, в котором максимальный из углов поворота между двумя дорогами из одного и того же города минимален.

Формат входного файла

Первая строка входного файла содержит целое чётное число N ($4 \leq N \leq 200$). Каждая из следующих N строк содержит 5 целых чисел. Первые два числа в i -й из этих строк — это X_i и Y_i ($-10^5 \leq X_i, Y_i \leq 10^5$). Следующие три числа — это номера городов, с которыми i -й город изначально соединен дорогами. Города нумеруются с единицы. Координаты никаких двух городов не совпадают.

Никакие 3 города не лежат на одной прямой линии. Каждый город соединен дорогами ровно с тремя городами. Между парой городов может быть не более одной дороги. Никакая дорога не соединяет город с самим собой. Любые два угла поворота (не обязательно при одном городе) в стране до реформы различаются не менее чем на 10^{-5} градуса. Любой угол поворота в стране до реформы отличается от угла в 60 градусов не менее чем на 10^{-5} градуса.

Формат выходного файла

Если поставленные королем условия выполнить невозможно, выведите единственную строку содержащую "Minister's life is short :(" (без крайних кавычек). Символы ", ':' и '(' имеют ASCII-коды 39, 58 и 40 соответственно.

Иначе выведите способ решения задачи, который следует выбрать министру, в виде N целых чисел, разделенных пробелами. Если i -е из выведенных чисел равно j , это означает, что министру следует убрать дорогу между городами i и j .

Пример

f.in	f.out
4 0 0 2 3 4 41 0 1 3 4 0 42 4 2 1 43 44 2 3 1	Minister's life is short :(
8 0 100 2 6 8 0 201 3 1 5 102 303 2 4 8 204 305 3 5 7 306 207 2 4 6 308 109 1 5 7 210 0 6 8 4 111 0 1 3 7	6 5 8 7 2 1 4 3

Задача G. Constellation (Division 1 Only!)

Имя входного файла: `g.in`
Имя выходного файла: `g.out`
Ограничение по времени: 3 seconds
Ограничение по памяти: 256 Mebibytes

Недавно исследователи обнаружили древний манускрипт, который описывает некоторое созвездие. Из него следует, что созвездие состоит из M звёзд, а также известны расстояния между каждой парой звёзд созвездия, если рассматривать небо как плоскость, а звёзды — как точки на этой плоскости.

На сегодняшнем небе в том полушарии, где предположительно находится описанное созвездие, видно N звёзд. Обычно созвездием считают объединение самых ярких звёзд какого-либо фрагмента неба, но за прошедшие тысячелетия яркости звёзд могли измениться, поэтому опираться на этот показатель уже невозможно. Следовательно, определять, какие звёзды сегодняшнего неба могут образовывать описанное в манускрипте созвездие, приходится лишь на основе данных расстояний.

Будем называть *возможным местоположением созвездия* список звёзд (I_1, I_2, \dots, I_M) такой, что для каждого i и j ($1 \leq i, j \leq M$) расстояние между звёздами I_i и I_j равно расстоянию между i -ой и j -ой звёздами манускрипта. Два возможных местоположения различны, если хотя бы на одной из позиций в них записаны разные звёзды.

Вам задано множество звёзд на сегодняшнем небе. Также дана матрица размерности $M \times M$, где элемент (i, j) обозначает квадрат расстояния между звёздами i и j созвездия. Подсчитайте количество возможных местоположений этого созвездия.

Формат входного файла

Первая строка входного файла содержит число M . Следующие M строк содержат по M целых чисел каждая — матрицу расстояний. Заданная матрица расстояний симметрична, её главная диагональ содержит только нули, а все остальные числа положительные и не превосходят 10^8 . Далее записано число N , а затем следует N строк, каждая из которых содержит пару целых чисел X_i, Y_i — координаты i -ой звезды на сегодняшнем небе ($2 \leq N \leq 3 \cdot 10^4$, $2 \leq M \leq \min(N, 20)$). Координаты каждой звезды — целые числа, не превосходящие 10^4 по абсолютному значению. Никакие две звезды не находятся в одной точке.

Формат выходного файла

Выведите количество возможных местоположений созвездия из манускрипта.

Пример

<code>g.in</code>	<code>g.out</code>
3 0 1 2 1 0 1 2 1 0 4 0 0 1 0 0 1 1 1	8

Note

Возможными местоположениями созвездия являются $(1,2,4)$, $(1,3,4)$, $(2,1,3)$, $(2,4,3)$, $(3,1,2)$, $(3,4,2)$, $(4,2,1)$, $(4,3,1)$.

Задача H. Presents (Division 1 Only!)

Имя входного файла: `h.in`
Имя выходного файла: `h.out`
Ограничение по времени: 4 seconds
Ограничение по памяти: 64 Mebibytes

Манао решил устроить вечеринку. Он ярко украсил свою хату, тщательно подобрал меню, созвал узкий круг друзей... и вдруг осознал, что забыл про подарки!

На своих вечеринках Манао всегда дарит гостям подарки. Более того, он всегда дарит каждому что-нибудь такое, чему тот будет рад. Сами гости тоже приносят подарки, каждый по одной штуке. Так как времени осталось немного, Манао не успеет закупить подарков, поэтому ему нужно найти способ выкрутиться из ситуации. И тогда его осеняет гениальная идея: перераспределить принесённые гостями подарки между ними! Естественно, отдать человеку принесённый им же подарок нельзя. Манао хорошо знает своих друзей и заранее может сказать, кому из них понравится принесённый кем подарок.

Но, к сожалению, может так случиться, что распределить подарки так, чтобы все друзья были рады, не получится. Поэтому Манао решил позвать вдобавок ещё некоторое количество знакомых. Манао хорошо знает и их, и поэтому может с уверенностью предсказать, кто что принесёт и чему будет рад. Обижать знакомых Манао тоже не хочет, поэтому каждый из них также должен получить подарок, которому будет рад. Но чтобы не превращать вечеринку в полный балаган, Манао пригласит наименьшее количество знакомых, которое обеспечит возможность распределить подарки между всеми гостями так, чтобы каждый был рад.

У Манао N друзей и M других знакомых. Пронумеруем друзей числами от 1 до N , а знакомых числами от $N + 1$ до $N + M$. Вам дана матрица, по которой можно определить, чей подарок кому понравится. Определите наименьшее количество знакомых, которых надо дополнительно позвать (все друзья уже приглашены). Если это невозможно, выведите число -1 .

Формат входного файла

Первая строка входного файла содержит два числа N и M ($2 \leq N \leq 100$, $0 \leq M \leq 100$). Каждая из следующих $N + M$ строк содержит по $N + M$ символов “Y” или “N”, причём j -ый символ в i -ой из этих строк обозначает, понравится ли человеку с номером i подарок человека с номером j . При этом ни одному гостю не может понравиться его собственный подарок.

Формат выходного файла

Выведите -1 , если никакое множество знакомых не спасёт вечеринку Манао, и размер такого минимального множества в противном случае.

Пример

h.in	h.out
2 2 NYYY NNYN YNNN NYNN	1
3 2 NNNYN YNYNN NNNYN NYNNY NNYNN	-1
3 2 NYNYN YNYNN NNNYN NYNNY NNYNN	2

Note

В первом примере можно пригласить человека с номером 3, а потом отдать гостю 2 подарок, принесённый гостем 3, гостю 3 подарок, принесённый гостем 1, а гостю 1 подарок гостя 2. В третьем примере можно, пригласив обоих знакомых, обменять подарки первого и второго друзей, четвертому отдать подарок третьего, третьему — пятого, а пятому — четвертого.

Задача I. ICPC (Division 1 Only!)

Имя входного файла:	<code>i.in</code>
Имя выходного файла:	<code>i.out</code>
Ограничение по времени:	4 seconds
Ограничение по памяти:	64 Mebibytes

Вот уже много лет существует мрачный культ ICPC. По свидетельствам очевидцев, служители этого культа занимаются очень странными вещами. Они разбиваются на группы по 3 человека и поклоняются экрану, от которого исходит таинственное свечение, с силой бьют по каким-то кнопкам, сгруппированным на непонятного рода панели. Коммуникация эта, видимо, не является однонаправленной — иногда “божество” подает какие-то знаки одобрения в ответ, и тогда раздаются дружные радостные крики служителей. У большинства нерадивых служителей, правда, ситуация эта происходит довольно редко. Чаще “божество” недовольно, что повергает поклоняющихся сначала в замешательство, а потом, нередко, и в уныние.

О природе “божества” ходят разные слухи. Некоторые утверждают, что “божество” существует во многих обличьях, каждое из которых управляется посвященным человеком (а может это и не человек вовсе, а существо более высокого порядка) — Админом. Особенную известность получил так называемый Админ с Большой Бородой. А еще с давних пор ходит слух об особой благосклонности “божества” к кефиру. . . Впрочем, это уже какая-то нелепица, да и от темы задачи мы отвлеклись.

А вот собственно и сама задача. Собрались как-то раз N служителей культа ICPC и решили, что традиционные методы поклонения устарели, надоели, да и вообще какие-то не особо зловецкие. Постановили они, что вместо усиленного ожесточенного стука по клавиатуре следует им обмениваться темной энергией. Был разработан план — множество из M пар (A, B) , означающих, что служитель с номером A должен передавать энергию служителю с номером B (соответственно, служитель с номером B должен получать энергию от служителя с номером A).

Начали они действовать согласно плану, но энергия почему-то не передавалась. Как вы сами понимаете, проблема ну никак не может быть в том, что нет у служителей ICPC никакой темной энергии! Понимали это и сами служители. Спустя некоторое время осознали они, что для того, чтобы план сработал, следует им расположиться определенным образом. Начертили они магический круг и расположились по его границам.

Конечно, одного наличия круга недостаточно, требуется еще, чтобы процесс передачи энергии был ассоциирован с кругом. Общеизвестно, что передача энергии ассоциирована с кругом, если каждый служитель передает энергию непосредственно следующим за ним в круге (при обходе круга по часовой стрелке) служителям и получает энергию от непосредственно следующих перед ним в круге служителей.

Определим то же самое более формально. Пусть $next(X)$ — номер служителя, стоящего непосредственно за служителем X по кругу (при обходе круга по часовой стрелке), а $prev(X)$ — номер служителя, стоящего непосредственно перед служителем X . Определим для $i > 1$ величины $next^i(X) = next(next^{i-1}(X))$ и $prev^i(X) = prev(prev^{i-1}(X))$ (будем также считать, что $next^1(X) = next(X)$ и $prev^1(X) = prev(X)$). Передача энергии ассоциирована с магическим кругом, если для каждого служителя X выполнены следующие условия:

- он передает энергию служителям $next^i(X)$, $1 \leq i \leq A$, где A — общее количество служителей, которым он передает энергию;
- он получает энергию от служителей $prev^i(X)$, $1 \leq i \leq B$, где B — общее количество служителей, от которых он получает энергию.

Никогда еще цель не была так близка, однако расположиться по кругу таким образом, чтобы достичь ассоциированной передачи энергии, у собравшихся служителей ICPC не получилось. Как братья по культу ICPC, вы, конечно же, приложите все усилия, чтобы помочь им!

Формат входного файла

Входной файл содержит несколько тестов. В первой строке файла записано их количество T , далее следует описание T тестов.

Описание каждого теста начинается со строки, содержащей числа N и M . Далее следует M строк, описывающих план передачи энергии. Каждая из этих строк содержит два целых числа — A и B . Служители пронумерованы числами от 1 до N ($N \geq 3$, $1 \leq A, B \leq N$, $A \neq B$). Все числа во входном файле целые, Сумма значений N по всем тестам в одном файле не превосходит 10^5 , сумма значений M по всем тестам в одном файле не превосходит $2 \cdot 10^5$. Также в рамках одного теста каждая пара (A, B) присутствует не более одного раза, и в рамках одного теста не могут одновременно присутствовать пары (A, B) и (B, A) .

Формат выходного файла

Для каждого теста во входном файле выведите строку, содержащую перестановку чисел от 1 до N — порядок, в котором следует стать служителям, чтобы достичь передачи энергии, ассоциированной с магическим кругом. Выведенный порядок должен соответствовать обходу круга по часовой стрелке. Если существует несколько решений, выведите любое из них. Если решения не существует, выведите “Epic fail” (без кавычек).

Пример

i.in	i.out
3	1 6 2 4 5 3
6 10	1 2 3
1 6	Epic fail
2 3	
2 4	
2 5	
3 1	
3 6	
4 3	
4 5	
5 3	
6 2	
3 0	
4 3	
1 2	
2 3	
3 1	

Задача J. Numbers (Division 2 Only!)

Имя входного файла: `j.in`
Имя выходного файла: `j.out`
Ограничение по времени: 1 second
Ограничение по памяти: 64 Mebibytes

Неотрицательное целое число называется K -значным, если его можно записать с помощью K цифр, а $(K - 1)$ цифр для этого недостаточно. Например, 43 является двузначным числом, 2010 — четырехзначное, а 0 и 5 — однозначные числа.

Для данных A и B подсчитайте, сколько существует K -значных неотрицательных целых чисел в десятичной системе счисления, где K не менее A и не более B .

Формат входного файла

Единственная строка содержит два целых числа A и B ($1 \leq A \leq B \leq 1000$).

Формат выходного файла

Выведите ответ задачи без ведущих нулей.

Пример

<code>j.in</code>	<code>j.out</code>
1 1	10
3 5	99900

Задача К. English (Division 2 Only!)

Имя входного файла: `k.in`
Имя выходного файла: `k.out`
Ограничение по времени: 1 second
Ограничение по памяти: 64 Mebibytes

В английском языке 26 букв, из которых 5 (а, е, і, о, u) — гласные, 20 согласных, а одну считают полугласной, т.е. её не считают ни гласной, ни согласной (буква ‘y’). Слово называется звучным, если в нём количество гласных превосходит количество согласных. Если слово содержит некоторую букву более одного раза, то каждое её вхождение засчитывается — то есть, например, слово “alabama” звучное.

Для данного слова найдите, какое наименьшее количество букв в нём надо заменить, чтобы получить звучное слово.

Формат входного файла

Единственная строка содержит данное слово. Длина данного слова не превосходит 1000 букв. Слово состоит из букв латинского алфавита в нижнем регистре.

Формат выходного файла

Выведите минимальное количество букв, при замене которых данное слово становится звучным.

Пример

<code>k.in</code>	<code>k.out</code>
dog	1
alabama	0
y	1

Задача L. Diamond (Division 2 Only!)

Имя входного файла:	1.in
Имя выходного файла:	1.out
Ограничение по времени:	1 second
Ограничение по памяти:	64 Mebibytes

Вы — большой любитель ювелирных украшений и драгоценных камней. Недавно в ювелирном магазине напротив Вашего дома в продаже появился совершенно замечательный бриллиант. Цена у него, однако, тоже была совершенно замечательная — X долларов. С сожалением Вам пришлось констатировать, что для Вас это слишком дорого, как бы ни нравился сам камень.

Видимо, другие потенциальные покупатели пришли к тому же выводу, потому что через некоторое время магазин устроил рекламную акцию. “Получите 5%-ную скидку на бриллиант за каждую букву Вашего имени!” — гласил транспарант у входа в магазин. Формулировка, прямо скажем, не очень однозначная, поэтому Вы решили поговорить с продавцами в магазине. В результате разговора выяснилось:

- Если буква встречается в имени несколько раз, то она будет подсчитана только единожды. Т.е., например, в имени alexandra с точки зрения данной акции не 9, а 7 букв, т.к. буква a встречается трижды.
- 5%-ная скидка умножается на количество различных букв в имени. Т.е. в случае имени alexandra будет предоставлена 35%-ная скидка.
- Скидки более 100% не предоставляются, но ровно 100%-ая скидка может быть предоставлена (когда Вы задавали вопрос по этому поводу, продавцы смотрели на Вас как на полоумного).

Конечно, таким щедрым предложением грешно было не воспользоваться, но хотелось сэкономить еще. По счастливой случайности, недавно Вы познакомились с человеком, подделывающим паспорта. Услуги его оплачиваются следующим образом:

- Подделка паспорта стоит A долларов. В поддельном паспорте можно использовать Ваше настоящее имя без изменений, а можно изменить его посредством применения сколь угодно большого количества операций каждого из указанных дальше трех видов.
- Вставка одной буквы в любое место имени стоит B долларов.
- Удаление одной любой буквы из имени стоит C долларов. Ситуация, когда в результате удаления получается пустая строка, считается недопустимой.
- Изменение одной любой буквы из имени на любую другую букву английского алфавита стоит D долларов.

По заданным числам X , A , B , C , D , а также Вашему настоящему имени $Name$, определите минимальную денежную сумму, которую придется потратить на покупку бриллианта (считая как деньги, заплаченные собственно за бриллиант, так и деньги, потраченные на подделку паспорта). Подделывать паспорт не обязательно, Вы также можете просто предъявить Ваш настоящий паспорт с именем $Name$ при покупке.

Формат входного файла

Первая строка входного файла содержит целое число X . Во второй строке записаны целые числа A , B , C и D , разделенные одиночными пробелами ($1 \leq X, A, B, C, D \leq 10^6$). В третьей строке записано $Name$ — строка размером от 1 до 15 символов, состоящая из латинских букв в нижнем регистре.

Формат выходного файла

Выведите минимальную денежную сумму, которую придется потратить на покупку бриллианта, измеренную в центах (в одном долларе 100 центов).

Пример

l.in	l.out
100 1 2 3 4 eldar	3100
100 100 100 100 100 ivan	8000
9876 1 567 890 1 nebuchadnezzar	296780

Задача M. Puzzle (Division 2 Only!)

Имя входного файла: `m.in`
Имя выходного файла: `m.out`
Ограничение по времени: 1 second
Ограничение по памяти: 64 Mebibytes

Задано N^2 десятичных цифр от 1 до 9.

Рассмотрим размещение этих цифр в клетках квадратной таблицы $N \times N$, по одной цифре в клетке. В каждой из строк таблицы, читая слева направо, получим десятичную запись некоторого N -значного числа. В каждом из столбцов таблицы, читая сверху вниз, также получим десятичную запись некоторого N -значного числа. Пусть S — это сумма всех N чисел по строкам и всех N чисел по столбцам.

Разместите числа внутри таблицы таким образом, чтобы значение S было максимальным.

Формат входного файла

Первая строка входного файла содержит целое число N ($1 \leq N \leq 8$). Во второй строке записаны без разделителей N^2 десятичных цифр от 1 до 9.

Формат выходного файла

Выведите максимально возможное значение суммы S .

Пример

<code>m.in</code>	<code>m.out</code>
2 9174	303