

Задача А. Просто поток

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

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.

Формат входных данных

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.

Формат выходных данных

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} .

Примеры

flow.in	flow.out
2	4
2	1
1 2 1	-3
2 1 3	

Задача В. Разрез

Имя входного файла: `cut.in`
Имя выходного файла: `cut.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

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

Формат входных данных

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.

Формат выходных данных

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).

Примеры

<code>cut.in</code>	<code>cut.out</code>
3 3 1 2 3 1 3 5 3 2 7	2 8 1 2

Задача С. Улиточки

Имя входного файла: `snails.in`
Имя выходного файла: `snails.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Две улиточки Маша и Петя сейчас находятся в на лужайке с абрикосами и хотят добраться до своего домика. Лужайки пронумерованы числами от 1 до n и соединены дорожками (может быть несколько дорожек соединяющих две лужайки, могут быть дорожки, соединяющие лужайку с собой же). В виду соображений гигиены, если по дорожке проползла улиточка, то вторая по той же дорожке уже ползти не может. Помогите Пете и Маше добраться до домика.

Формат входных данных

В первой строке файла записаны четыре целых числа — n , m , s и t (количество лужаек, количество дорог, номер лужайки с абрикосами и номер домика). В следующих m строках записаны пары чисел. Пара чисел (x, y) означает, что есть дорожка с лужайки x до лужайки y (из-за особенностей улиток и местности дорожки односторонние). Ограничения: $2 \leq n \leq 10^5$, $0 \leq m \leq 10^5$, $s \neq t$.

Формат выходных данных

Если существует решение, то выведите YES и на двух отдельных строчках сначала последовательность лужаек для Машеньки (дам нужно пропускать вперед), затем путь для Пети. Если решения не существует, выведите NO. Если решений несколько, выведите любое.

Примеры

<code>snails.in</code>	<code>snails.out</code>
3 3 1 3	YES
1 2	1 3
1 3	1 2 3
2 3	

Замечание

Дан оргграф, найти два непересекающихся по ребрам пути из s в t , вывести вершины найденных путей.

Задача D. Великая стена

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

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.

Формат входных данных

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 .

Формат выходных данных

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 .

Примеры

стандартный ввод	стандартный вывод
5 5 --... A-.#- .#.#- --.-- --.-B	3 --...+ A-+#- +#. #- --.-- --.-B
1 2 AB	-1
2 2 A# #B	0 A# #B

Задача Е. Космические перевозки

Имя входного файла: `bring.in`
Имя выходного файла: `bring.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

К 3141 году человеческая цивилизация распространилась по всей галактике. Для перехода от одной звездной системы к другой используются специальные гипертуннели. Чтобы использовать гипертуннель, нужно прилететь в специальное место рядом с исходной звездой используя ваш космический корабль, активировать гипердрайв, пролететь через гипертуннель, выйти рядом со звездой назначения и лететь на нужную вам планету. Весь процесс занимает ровно один день. Небольшой недостаток системы состоит в том, что по каждому туннелю может пролететь только один космический корабль в день.

Вы работаете в транспортном отделе компании «Intergalaxy Business Machines». Сегодня утром ваш начальник дал вам новую задачу. Чтобы запустить чемпионат по программированию IBM, нужно доставить K суперкомпьютеров от Земли, где находится штаб-квартира компании на планету Эйсиэм.

Поскольку суперкомпьютеры очень большие, нужно, для перевозки одного нужен целый космический корабль. Вас попросили найти план доставки суперкомпьютеров, который позволит доставить все компьютеры за минимальное число дней. Поскольку IBM является очень мощной корпорацией, можете считать, что каждый раз, когда вам нужен какой-то гипертуннель, он к вашим услугам. Однако вы все равно можно использовать каждый туннель только один раз в день.

Формат входных данных

Первая строка входного файла содержит N — число звездных систем в галактике, M — число туннелей, K — число суперкомпьютеров для доставки, S — номер солнечной системы, где находится планета Земля и T — номер звездной системы, где находится Планета Эйсиэм ($2 \leq N \leq 50$, $1 \leq M \leq 200$, $1 \leq K \leq 50$, $1 \leq S, T \leq N$, $S \neq T$)

Следующие M строк содержат по два разных целых числа и описывают туннели. Для каждого туннеля даются номера звездных систем, которые он соединяет. По туннелю можно путешествовать в обоих направлениях, но помните, что каждый день только один корабль может использовать туннель. В частности, два судна не могут одновременно проходить через один и тот же туннель в противоположных направлениях. Ни один туннель не соединяет звезду с самим собой и две звезды связаны не более чем одним туннелем.

Формат выходных данных

В первой строке выходного файла выведите L — наименьшее число дней, необходимых для доставки K суперкомпьютеров от звездной системы S до звездной системы T с использованием гипертуннелей.

Следующие L строк должны описывать процесс. Каждая строка должна начинаться с C_i — числа кораблей, которые отправляются из одной системы в другую в этот день. Далее должны следовать C_i пар целых чисел, пара A, B означает, что корабль перемещается из звездной системы A в звездную систему B .

Пример

bring.in	bring.out
6 7 4 1 6	4
1 2	2 3 2 4 4
2 3	3 2 4 3 3 4 6
3 5	3 1 4 2 6 3 5
5 6	2 1 6 3 6
1 4	
4 6	
4 3	

Задача F. Групповой турнир

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

В нашем капиталистическом и меркантильном мире всё решают деньги, и даже спорт не стал исключением. Все команды-участницы уже купили себе нужное количество очков в следующем сезоне, и местной федерации хоккея осталось только распределить результаты предстоящих игр. Однако, некоторые команды не поскупились и помимо покупки очков также купили ещё и результаты некоторых игр. Поначалу в федерации думали, что это им только упростит задачу: чем для большего числа игр результаты уже определены, тем меньше работы остаётся им. Но позже они поняли, что ошиблись. Они попросили вас стать участником их коррупционной схемы и помочь с распределением результатов игр предстоящего сезона.

Местный хоккейный турнир проходит по круговой системе: в турнире участвуют N команд и каждая команда играет с каждой ровно одну игру. За игру команды получают очки по следующим правилам:

- Если победителя удалось выявить в основное время матча, то ему достаётся 3 очка, а проигравшему — 0.
- Если основное время закончилось вничью и для выявления победителя понадобилось дополнительное время (овертайм), то победителю дают 2 очка, а проигравшему — 1 очко. Овертайм не ограничен во времени и длится до тех пор, пока одна из команд не забьёт гол.

По итогам турнира очки команды определяются как сумма её очков по всем сыгранным играм.

Формат входных данных

В первой строке входного файла содержится целое число N — количество участников турнира ($2 \leq N \leq 100$). Команды занумерованы числами от 1 до N .

Следующие N строк файла содержат по N символов и представляют собой турнирную таблицу на данный момент. Символ a_{ij} в строке i ($1 \leq i \leq N$) на позиции j ($1 \leq j \leq N$) означает результат игры команды номер i с командой номер j и может быть одним из:

- 'W' — означает, что команда i обыграет команду j в основное время матча;
- 'w' — команда i обыграет команду j в овертайме;
- 'L' — команда i проиграет команде j в овертайме;
- 'l' — команда i проиграет команде j в основное время матча;
- '.' — если результат игры между командами i и j ещё не определён;
- '#' — если i равно j , означает отсутствие данного матча, т. к. команда не может играть сама с собой.

Гарантируется, что данная таблица корректна. Более формально:

- $a_{ij} = \text{'\#'}$ для всех $i = j$;
- если $a_{ij} = \text{'.'}$, то $a_{ji} = \text{'.'}$;
- $a_{ij} = \text{'W'}$ тогда и только тогда, когда $a_{ji} = \text{'L'}$;
- $a_{ij} = \text{'w'}$ тогда и только тогда, когда $a_{ji} = \text{'l'}$.

Последняя строка входного файла содержит N целых чисел p_i — количество очков, которое требуется набрать i -й команде ($1 \leq i \leq N$).

Формат выходных данных

В выходной файл выведите полностью заполненную турнирную таблицу в формате, аналогичном формату входного файла.

Гарантируется, что решение существует. Если решений несколько, то можно вывести любое из них.

Пример

стандартный ввод	стандартный вывод
4	#wWw
#..W	l#wW
.#w.	Ll#w
.l#.	LlL#
L..#	
8 6 3 1	

Задача G. Тараканы

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

Несколько чистоплотных тараканьих семей хотят поселиться в однокомнатной квартире. Квартира состоит из большой комнаты и кухни, соединённых узким коридором. Тараканов совершенно не интересует коридор сам по себе, однако они хотят иметь доступ и к кухне, и к комнате. Для того чтобы тараканья семья имела такой доступ, ей нужен индивидуальный транспортный путь через коридор. Тараканы — умные насекомые, поэтому они решили составить план коридора и определить, какое максимальное количество транспортных путей можно проложить через коридор.

В тараканьем плане коридор представляется бесконечной в обе стороны полосой шириной W сантиметров. Тараканы начертили прямоугольную систему координат, ось X которой параллельна направлению коридора. Хозяйева квартиры расставили в коридоре некоторое количество массивных предметов. Каждый предмет является прямоугольником со сторонами, параллельными осям координат, и вершинами с целыми координатами в сантиметрах. Границы коридора задаются уравнениями $y = 0$ и $y = W$. На координатной плоскости тараканы нарисовали квадратную сетку со стороной одной клетки, равной 1 см, начиная от границы коридора.

Тараканы договорились, что каждый транспортный путь является бесконечной в обе стороны цепью квадратных клеток. Цивилизованные тараканы никогда не прыгают, поэтому две подряд идущие клетки в этой цепи должны быть соседними по стороне. Цивилизованные тараканы всегда бегают по полу, поэтому никакая клетка цепи не может пересекаться с хозяйской вещью или выходить за пределы коридора. Тараканы не любят бегать по кругу, поэтому все клетки цепи различны. Тараканы не хотят давки в час пик, поэтому они требуют, чтобы различные транспортные пути не пересекались, то есть не имели общих клеток. Транспортный путь должен соединять кухню и комнату, поэтому разные концы цепи должны уходить в разные стороны коридора.

Ваша задача — определить максимальное количество транспортных путей, которые можно проложить через коридор.

Формат входных данных

В первой строке входного файла записано два целых числа N и W . N — количество предметов в коридоре, W — ширина коридора в сантиметрах ($0 \leq N \leq 5000$, $0 < W \leq 10^9$).

Каждая из последующих N строк описывает одну из хозяйских вещей. Она содержит четыре целых числа X_1, Y_1, X_2, Y_2 — координаты двух противоположных углов прямоугольника ($-10^9 \leq X_1 < X_2 \leq 10^9$, $0 \leq Y_1 < Y_2 \leq W$). Предметы, расставленные в коридоре, могут пересекаться.

Формат выходных данных

В выходной файл нужно вывести одно целое число — максимально возможное количество транспортных путей.

Пример

стандартный ввод	стандартный вывод
2 9 -4 4 -1 7 2 1 5 5	5

Задача Н. Два языка

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

Формат входных данных

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.

Формат выходных данных

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

Пример

стандартный ввод
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
стандартный вывод
1 4 3 8

Задача I. Восстановление бактерии

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

У несчастного ученого Джона большие проблемы на работе. Коллега дал ему чашку Петри и попросил его посчитать количество различных видов бактерий, которые там живут. Для этого Джон принес чашку в лабораторию секвенирования. К сожалению, оборудование довольно старое, и оно возвращает только список различных k -меров геномов бактерий, которые находятся в чашке.

Он решил, что он может оценить количество различных видов бактерий в чаше как минимальное количество геномов бактерий, таких что каждый k -мер встречается, по крайней мере, в одном геноме, и геномы не содержат k -меров, кроме предоставленных. Помогите ему найти это число!

В этой задаче мы рассматриваем цепочку ДНК бактерий как линейную (не циклическую), а k -меры берутся только в одном направлении.

Формат входных данных

Первая строка содержит два целых числа n и k ($1 \leq n, k \leq 100$) — число различных k -меров, которые были прочитаны и их длина.

Каждая из следующих n строк содержит один k -мер — строку длины k , состоящую из букв А, С, G и Т.

Формат выходных данных

Первая строка ответа должна содержать m — минимальное количество различных видов бактерий. Следующие m строк должны содержать последовательности геномов этих бактерий. Каждая последовательность должна быть не меньше k и не больше 10000 символов.

Примеры

стандартный ввод	стандартный вывод
4 2 AT TA GC CG	2 TAT CGC
5 3 ACT TCT CTA TAT TAG	2 TCTAT ACTAG
1 1 A	1 A