

Задача А. Имя кота

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

Никита взял из приюта котёнка и теперь задумался, как бы ему котёнка назвать. Посмотрев, передачу про кошек, Никита узнал, что кошки хорошо воспринимают звуки «к» и «с». Тогда он решил определять хорошестть имени котёнка следующим образом:

- Имя котёнка записывается при помощи строчных латинских букв.
- Никита считает количество букв «к», количество букв «s» и количество раз, которое встречается сочетание «ks» в имени.
- Сумма этих количеств и является хорошестью имени котёнка.

Вам дано имя котёнка, определите его хорошестть по Никитиным правилам.

Формат входного файла

В первой строке входного файла находится строка, длиной не более 100 символов, состоящая из маленьких латинских букв — имя котёнка, которое придумал Никита.

Формат выходного файла

Выведите единственное число — хорошестть имени.

Примеры

<code>catname.in</code>	<code>catname.out</code>
<code>kokos</code>	3
<code>piksel</code>	3
<code>kso</code>	3

Задача В. Работа, кофе, профит

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

Программист Саша очень любит свою работу. Каждый будний день, он просыпается и начинает свой день с кружки горячего молотого кофе, после чего он получает необходимый заряд бодрости, одевает тапочки и идет реализовывать свои идеи в виде кода. Но спустя некоторое время, он опять испытывает необходимость в очередной кружке кофе, чтобы снова приступить к своей работе...

Попробуем математически описать поведение Саши. Проснувшись после длинной (или короткой) ночи, Саша обладает зарядом бодрости Q . Будем считать, что весь рабочий день поделен на N промежутков времени, в каждый из которых он может либо работать, либо пить кофе. Если в i -ый промежуток времени Саша работает, то:

- Саша теряет заряд бодрости q_i (если у Саши было менее q_i единиц бодрости, то его заряд бодрости становится равным нулю);
- завершенность продукта увеличивается на p_i единиц завершенности продукта (PROduct FINishness uniTs, PROFIT);
- если у Саши было меньше q_i единиц бодрости, то следующие K промежутков времени Саша ничего не сможет делать, кроме как пить кофе, общаться с друзьями в социальных сетях или играть в какую-нибудь игру на своей приставке;
- ??????
- PROFIT

Если в i -тый промежуток времени Саша пьет кофе, то восстанавливает R единиц бодрости. Однако, заряд бодрости Саши не может стать больше чем 100 единиц.

Саша просит Вас узнать, сколько единиц PROFIT он сможет получить за день, если будет действовать оптимальным образом.

Формат входного файла

Первая строка входного файла содержит целые числа Q, N, K, R ($0 \leq Q \leq 100, 1 \leq N \leq 100, 1 \leq K \leq N, 1 \leq R \leq 100$). В следующих N строках содержатся два целых числа q_i, p_i ($1 \leq q_i \leq 100, 1 \leq p_i \leq 10000$).

Формат выходного файла

Выведите единственное число — максимальное количество единиц PROFIT, которое может получить Саша.

Примеры

<code>coffee.in</code>	<code>coffee.out</code>
10 8 5 2 55 6 6 1 88 3 31 3 54 7 16 18 71 3 28 9	27

Задача С. Разрез торта

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

У Александра сегодня гости — он отмечает день рождения. Естественно не обошлось без песен, плясок и торта. Все шло просто прекрасно и казалось, что ничто не предвещало проблем, но Нежданчик всегда появляется внезапно. Нежданчик — это друг Александра. Он большой пессимист, и всё время говорит Александру, что у него ничего не получится. Вот и сегодня он заявил, что торт не получится разделить на всех гостей так, чтобы никто не был обижен.

Теперь у Александра возникла задача: проверить, действительно ли торта не хватит. Присмотревшись к торту, он заметил, что длина у торта бесконечна, а вот ширина оставляет желать лучшего, поэтому стоит резать торт параллельными прямыми, вдоль длины торта (Александр в себе более чем уверен, поэтому считает, что он сможет разрезать торт). Но поскольку он не очень аккуратен, то его разрезы имеют ширину T .

Более формально: торт представляет из себя полосу ширины L , разрезы представляют из себя полосы ширины T . Пообщавшись с гостями, Александр получил много хороших впечатлений, а еще он узнал, что i -й гость желает отведать полоску торта шириной не менее a_i . Помогите Александру определить, хватит ли торта, что бы каждому гостю выдать полоску той ширины, которую он желает?

Формат входного файла

В первой строке содержатся три целых числа $1 \leq N \leq 10^5$ — количество гостей, $1 \leq L \leq 10^9$ — ширина торта и $1 \leq T \leq 10^9$ — ширина разреза. Во второй строке содержится N чисел $1 \leq a_i \leq 10^9$ — ширина куска, которую хочет i -й гость.

Формат выходного файла

Выведите "YES" (без кавычек), если разрезать торт по указанным правилам возможно и "NO" иначе.

Примеры

<code>cut.in</code>	<code>cut.out</code>
2 6 2 1 2	YES
2 3 2 1 2	NO

Задача D. Игра

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

Антон в свободное от учебы время любит играть в различные игры наподобие Mario. Недавно он нашел упрощенный вариант этой игры.

- Игровое поле состоит из системы координат, ось OX параллельна поверхности Земли (в таких масштабах считаем Землю плоской), ось OY перпендикулярна поверхности Земли и направлена от неё, следовательно сила тяжести направлена вертикально вниз (в направлении, противоположном оси OY), других сил на игровом поле нет.
- В данной игре считаем героя игры материальной точкой.
- Поле содержит платформы — горизонтальные отрезки, расположенные фиксированно, по которым игрок может свободно перемещаться и на которых можно стоять, не боясь упасть.
- Герой может прыгать, мгновенно задавая некоторую начальную скорость и угол. При приземлении на платформу, он так же мгновенно останавливается, не отскакивая и не улетая за пределы платформы.
- В полёте, герой может использовать напиток `Hitriy`, выпив который он сможет пролететь насквозь ровно одну платформу, не остановившись на ней. В течение одного полёта герой может использовать напиток сколько угодно раз. Это позволяет ему выбирать, на какой платформе остановить своё движение.
- Поскольку герой не супермен, он ограничен в начально задаваемой скорости прыжка.
- Цель игры — добраться от начальной платформы до конечной путём прыжков и перемещений, ни разу не упав.

Антон не особо то любит программировать и просит Вас по заданному положению платформ, игрока, желаемой финальной платформы и максимальной допустимой начальной скорости прыжка определить, сможет ли герой игры добраться до конечной платформы.

Формат входного файла

В первой строке входного файла содержатся целые числа n , V , s , f ($1 \leq n \leq 1000, 1 \leq V \leq 1000, 1 \leq s, f \leq n$) — число платформ на игровом поле, максимальная допустимая стартовая скорость прыжка, начальная платформа игрока и финальная платформа. Следующие n строк содержат описания платформ: i -ая из этих строк содержит три числа $y_i, x1_i, x2_i$ — y -координата платформы и x -координаты концов платформы ($-10^6 \leq x1_i < x2_i \leq 10^6, |y_i| \leq 10^6$). Все координаты заданы в метрах, считайте, что ускорение свободного падения: $g = 9.8\text{м/с}^2$.

Формат выходного файла

В единственной строке файла выведите ‘YES’, если герой игры сможет добраться до финальной платформы, и — ‘NO’ иначе.

Примеры

<code>game.in</code>	<code>game.out</code>
2 4 1 2 0 -6 10 0 -9 -7	YES

Задача Е. Пистолет

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

Александр очень любит свой пистолет. А еще он любит петь песни. Но пистолет он любит больше, поэтому, даже когда Александр поёт, он не забывает про свой пистолет. Поскольку концентрироваться на пистолете во время пения песен невозможно, Александр придумал как не обидеть пистолет и не сбиться с ритма: он крутит пистолет.

Более точно, пистолет представляет из себя отрезок с закрепленным концом, вокруг которого он вращается. Но вращать пистолет не так и просто, особенно пистолет Александра, поэтому скорость вращения пистолета не постоянна. В разные промежутки времени скорость вращения разная. Александр даже заметил, что во время исполнения его любимой песни про битву гладиаторов, есть N промежутков времени, длина i -го промежутка — t_i секунд, а угловая скорость вращения на i -м промежутке — w_i радиан в секунду.

В конце дня Александр чистит пистолет. Для этого ему нужно узнать, какое расстояние прошел конец пистолета во время исполнения его любимой песни. Длина ствола у пистолета — L метров. Помогите Александру посчитать, какое расстояние прошел конец его пистолета сегодня во время исполнения его любимой песни

Формат входного файла

В первой строке содержатся два целых числа $1 \leq N \leq 1000$ — количество промежутков и $1 \leq L \leq 100$ длина пистолета в метрах. В последующих N строках содержится по два целых числа $1 \leq w_i \leq 100$ — скорость вращения в радианах в секунду и $1 \leq t_i \leq 100$ — длина промежутка в секундах.

Формат выходного файла

Выведите единственное число — расстояние пройденное концом пистолета в метрах.

Примеры

<code>gun.in</code>	<code>gun.out</code>
1 2 3 1	6

Задача F. K -инверсии

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

Никита изучает комбинаторику. Недавно он познакомился с такими забавными объектами, как перестановки. Напомним, что перестановкой чисел от 1 до n называется последовательность чисел от 1 до n в каком-то порядке, в которой каждое из этих чисел встречается ровно один раз.

Также Никите рассказали про инверсии. Инверсия — это такая ситуация, когда в перестановке большее число идёт перед меньшим. Например, в перестановке 2 1 3 есть инверсия из чисел 2 и 1.

А недавно Никита узнал про k -инверсии. Это такие ситуации, когда одно число в перестановке идёт раньше другого, и при этом первое больше второго не менее, чем на k . Таким образом, обычные инверсии являются 1-инверсиями. Более формально, k -инверсия — это пара индексов (i, j) таких, что $i < j$ и $p_i \geq p_j + k$.

Никита просит Вас посчитать количество k -инверсий в заданной перестановке.

Формат входного файла

Первая строка входного файла содержит числа n и k . ($1 \leq k \leq n \leq 10^5$) Во второй строке находится n чисел — перестановка p .

Формат выходного файла

Выведите единственное число — количество k -инверсий в заданной перестановке.

Примеры

<code>k-inversions.in</code>	<code>k-inversions.out</code>
3 1 2 1 3	1
3 2 2 1 3	0
3 2 3 2 1	1

Задача G. Пенобетон

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

Денис и Вова строят загородный дом. Уже залит фундамент, теперь они переходят к возведению стен. В качестве материала стен Денис выбрал пенобетон. Стоит отметить ряд полезных характеристик этого материала.

- Пенобетон обладает намного лучшими теплоизоляционными свойствами, чем обычный бетон.
- На производство пенобетонного изделия требуется в 2-4 раза меньше цемента.
- Пенобетонное изделие имеет меньшую по сравнению с бетонным массу, что снижает расходы на транспортировку, кладку и обработку.
- Экологическая чистота аналогична бетону. При производстве пеноблока используются только цемент, песок и вода.
- Пенобетон достаточно гидроустойчив.
- Пенобетон по простоте обработки сравним с деревом: он легко пилится, сверлится, гвоздится.

Пеноблок — это строительный блок, получаемый из пенобетона. Он представляет из себя прямоугольный параллелепипед.

Денис уже заранее создал макет дома, определился с высотой и длиной стен. Так как стройка — дело затратное, было решено заранее высчитать, какое минимальное число пеноблоков потребуется для возведения каждой стены.

Стоит отметить вышеприведенное качество пенобетона — возможность очень легко и быстро резать пеноблок на части. Денис уже купил резак для пенобетона. Он может разделять пеноблок на части. Цена деления шкалы резака — 1 метр (т.е. в результате разрезания получаются части с шириной равной целому числу метров). От разрезания, остается шершавая поверхность. Получается, что только две части будут иметь одну гладкую сторону и одну шершавую. Остальные части будут иметь по две шершавых стороны.

Так случилось, что Вова не только хороший помощник в строительстве, но также увлекается математикой и программированием. Собрав нужную информацию, он, как любой математик, сформулировал задачу.

Есть стена шириной B метров. Стена должна состоять из N рядов пеноблоков. Каждый пеноблок идентичен друг другу и имеет ширину A метров.

В итоге должна получиться стена глубиной один пеноблок, шириной B метров, высотой N пеноблоков и на её возведение должно быть потрачено минимальное количество пеноблоков. В готовой стене, все стороны должны быть гладкими (т.е. по краям ряда допускается размещать только части пеноблока с гладкой стороной, внутри же стены можно использовать части с шершавыми сторонами).

Помогите Денису проверить правильность расчётов Вовы.

Формат входного файла

Во входной файле заданы три целых числа, разделенных пробелом — A , B , N ($1 \leq A, B, N \leq 100, A \leq B$).

Формат выходного файла

В выходной файл выведите одно единственное целое число — минимальное количество пеноблоков, необходимых для возведения стены.

Примеры

penobeton.in	penobeton.out
2 3 1	2
1 3 2	6

Задача N. PiCell

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

Вы являетесь сотрудником отдела разработки компании PiSystems. В целях повышения программной безопасности, отделу было поручено написать свою собственную версию Microsoft Excel. Данный проект получил кодовое название PiCell. Менеджер проекта распределил обязанности и Вам выпала задача реализации алгоритма изменения формул при копировании из ячейки в ячейку.

PiCell задумывалась как простая система для работы с табличными данными, поэтому было решено, что формулы в этой системе могут представлять из себя лишь простейшие математические выражения, содержащие только ссылки на ячейки, целые числа, знаки $+$, $-$, $*$, $/$ и круглые скобки.

Диапазон ячеек в PiCell ограничен размерами 18277 на 99999 по столбцам и строкам, соответственно. При этом, как и в любом табличном редакторе, столбцы принято нумеровать буквами латинского алфавита. Номер столбца представляет из себя десятичное число, переведенное в символьную систему счисления, где каждая цифра кодируется символами латинского алфавита ($0=A$, $1=B$, ..., $24=Y$, $25=Z$, $26=AA$, $27=AB$, ..., $700=ZY$, $701=ZZ$, $702=AAA$, $703=AAB$, ..., $18276=ZZY$, $18277=ZZZ$). Более формально, столбцы от нуля до 25 кодируются одной буквой латинского алфавита, столбцы с номерами от 26 до 701 включительно — двумя буквами, столбцы от 702 до 18277 — тремя буквами.

Ячейка записывается как конкатенация номера столбца и строки. Минимальная ячейка — $A1$, максимальная — $ZZZ99999$.

Ссылка на ячейку бывает трех типов:

1. простая ссылка, которая содержит динамический столбец и строку (при копировании формулы, если координаты конечной ячейки отличаются от начальной на i по столбцам и j по строкам, то номер столбца у ссылки в формуле изменится на i , а номер строки — на j);
2. полустатичная ссылка, которая содержит динамический столбец и статичную строку или статичный столбец и динамическую строку (при копировании формулы, статичное значение строки или столбца не изменяется);
3. статичная ссылка, которая содержит статичный столбец и строку (при копировании формулы ячейка вообще не изменяется).

Стоит отметить, что если при копировании формулы из ячейки в ячейку, какая-то из ссылок будет иметь не допустимый ограничениями номер строки или столбца (т.е. номер строки будет меньше 1 или больше 99999, или номер столбца в десятичной системе будет меньше 0 или больше 18277), то этот номер следует округлить до ближайшего допустимого значения. Таким образом, ссылки внутри формулы никогда не могут указывать на область вне диапазона ячеек программы PiCell.

Вам даётся начальная ячейка, конечная ячейка и формула, записанная в начальной ячейке. Требуется разработать алгоритм для PiCell, который произведет замену ссылок внутри формулы, с учётом сдвига из начальной ячейки в конечную, оставив все остальные символы внутри формулы неизменными.

Формат входного файла

В первой строке входного файла находятся координаты начальной ячейки в записанные как AB , где A — это номер столбца записанный с помощью прописных латинских символов, а B — номер строки. Во второй строке входного файла находятся координаты конечной ячейки в записанные как AB . В третьей строке входного файла находится формула из начальной ячейки. Формула всегда не пуста, представляет из себя строку длиной не более 1000 символов, содержащую только символы $+$, $-$, $*$, $/$, круглые скобки и ссылки на ячейки. В формуле отсутствуют любые пробельные символы.

Простая ссылка на ячейку представляет из себя запись CD , где C — это номер столбца, записанный с помощью прописных латинских символов, а D — номер строки. Полустатичная ссылка на ячейку представляет из себя запись $\$CD$ или $C\$D$, где символ $\$$ обозначает статичность столбца или строки. Статичная ссылка на на ячейку представляет из себя запись $\$C\D .

Формат выходного файла

В выходной файл выведите вид формулы, получившийся после её переноса из начальной ячейки в конечную.

Примеры

picell.in	picell.out
A1 B3 A1+A2-C4	B3+B4-D6
IP53 DK68 CJ30/\$RS\$20	A45/\$RS\$20
N033 ZU31 UT27+LE91/\$WY2/(CM\$90)	AGZ25+XK89/\$WY1/(OS\$90)
I5 X5 \$I\$2*(\$H5-(\$D1*(B5-(R8)-S\$4+(B\$6))))	\$I\$2*(\$H5-(\$D1*(Q5-(AG8)-AH\$4+(Q\$6))))

Задача I. Последовательности

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

Саша изучает последовательности. Сегодня он занимается последовательностями из n чисел, каждое из которых может быть от 1 до c . Саше нравятся последовательности, в которых много одинаковых чисел идёт подряд. Поэтому для каждой последовательности $a : a_1, a_2, \dots, a_n, 1 \leq a_i \leq c$ Саша вычисляет $f(a)$ — максимально количество подряд идущих одинаковых чисел.

Например если $n = 3, c = 2$, то для последовательности $1\ 2\ 2$ $f(a) = 2$, а для последовательности $1\ 2\ 1$ $f(a) = 1$.

Сашу интересует сумма $f(a)$ по всем возможным последовательностям. Так как сумма может быть очень большой, он просит посчитать её по модулю $10^9 + 7$.

Формат входного файла

В первой строке входного файла находятся два числа n и c — длина последовательности и максимальное значение её элемента соответственно. ($1 \leq n \leq 1000, 1 \leq c \leq 10^9$)

Формат выходного файла

Выведите сумму $f(a)$ по всем последовательностям, по модулю $10^9 + 7$.

Примеры

<code>sequence.in</code>	<code>sequence.out</code>
2 4	20
3 2	16

Задача J. Похожие числа

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

Вам дано два числа A и B . Необходимо найти такие системы счисления с основанием X и Y , что A в системе счисления с основанием X записывается так же, как и B — в основании Y . В данной задаче предполагается существование систем счисления с основаниями от 2 до 36. Цифры со значениями выше 9 кодируются символами латинского алфавита.

Формат входного файла

В первой строке входного файла содержится два целых числа A , B , записанные в десятичной системе счисления ($1 \leq A, B \leq 10^9$).

Формат выходного файла

В единственной строке файла выведите два числа X и Y , разделяя их пробелом, или — “-1”, если таких X и Y не существует.

Примеры

<code>similar.in</code>	<code>similar.out</code>
42 27	12 7

Примечание

В системе счисления с основанием 7, число 27 запишется как 36. Точно так же выглядит запись числа 42 в системе счисления с основанием 12.