

Problem A. Again About Distancing

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 512 mebibytes

Назовём **безопасной** строку, состоящую из строчных латинских букв и пробелов, в которой между любыми двумя буквами расположены не менее двух пробелов.

Вам дана строка, проверьте, является ли она безопасной.

Input

Входные данные содержат строку, состоящую из не более чем 256 символов. Гарантируется, что строка состоит из строчных латинских букв, не начинается с пробела и не заканчивается пробелом.

Output

Выведите **safe**, если строка является безопасной, и **unsafe** в противном случае.

Examples

standard input	
s	o c i a l d i s t a n c i n g
standard output	
safe	
standard input	
virus	
standard output	
unsafe	

Problem B. Bad N -set

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 512 mebibytes

Назовём **плохим N -набором** набор из N отрезков, обладающий следующим свойством.

- Длина каждого отрезка является целым положительным числом, причём среди отрезков могут быть и совпадающие.
- Из набора нельзя выбрать какое-то количество отрезков так, чтобы из них можно было сложить многоугольник ненулевой площади

По заданному N определите наименьшую суммарную длину отрезков из плохого N -набора.

Input

Первая строка входных данных содержит одно целое число N ($1 \leq N \leq 42$).

Output

Выведите одно целое число — ответ к задаче.

Examples

standard input	standard output
1	1
2	2

Problem C. Card Game

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 512 mebibytes

В казино Байтландии появилась новая игра, называемая «Математический блэкджек».

Игра идёт колодой из 52 карт, по 13 карт в каждой масти. Каждой карте внутри данной масти присваивается числовое значение от 1 до 13 (то есть туз имеет значение 1, двойка — 2... валет — 11, дама — 12, король — 13). Игрокам раздаётся по четыре карты. На этапе подсчёта очков разрешается строить из значений своих карт арифметические выражения, используя сложение, умножение, вычитание, деление и скобки. Карты могут следовать в выражении в любом порядке, использовать надо все четыре карты ровно по одному разу.

Значение выражения должно оказаться **целым** числом, как можно более близким к 21. Хотя понятия «перебор» в этом варианте блэкджека и нет, в случае равного расстояния до 21 выигрывает меньшее число, то есть, например, при возможности построения 23 и 19 надо выбирать 19.

По заданным четырём картам постройте выражение с наилучшим в смысле вышеописанных правил значением.

Input

В первой строке входа заданы четыре целых положительных числа, не превосходящих 13 — номиналы разданных Вам на руки карт.

Output

Одно целое число — ответ на задачу.

Examples

standard input	standard output
2 4 3 1	21
4 4 4 3	20

Note

В первом примере $1 + (2 + 3) \cdot 4 = 21$. Во втором примере 21 не достигается. И, хотя $(4 \cdot 4 \cdot 4) / 3 = 21.(3)$, это значение не является целым, так что ответом является 20 ($4 \cdot (4 + 4 - 3) = 20$).

Problem D. Damaged Bridge

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 512 mebibytes

...После наводнения железнодорожный мост оказался сильно повреждён. Провоз по мосту более чем N тонн груза одновременно мог привести к обрушению моста. К счастью, это удалось выяснить до того, как по мосту прошёл поезд... но позже того, как поезд отправился.

Машинист принял решение отцепить некоторое число вагонов таким образом, чтобы оставшаяся часть вместе с локомотивом смогла беспрепятственно преодолеть мост. Длина моста равна 60 метров, длина локомотива и каждого из вагонов — по 15 метров, вес локомотива и каждого вагона в тоннах задан.

Уже проехавшие или ещё не заехавшие на мост вагоны влияния на нагрузку не оказывают, равно как и вагоны, находящиеся на мосту не полностью. Требуется выяснить, какое наибольшее количество вагонов можно оставить в головной части поезда

Input

В первой строке ввода задано одно целое число N — максимальная нагрузка на повреждённый мост в тоннах ($1 \leq N \leq 10^6$). В следующей строке задано целое число M ($1 \leq M \leq 10^6$) — количество вагонов поезда (с учётом локомотива). В следующей строке задана масса локомотива w_0 в тоннах, в i -й из последующих $M - 1$ строк — масса i -го вагона w_i в тоннах (все w_i целые, $1 \leq w_i \leq 10^6$).

Output

Выведите одно целое число — наибольшее количество вагонов, которое можно оставить в головной части эшелона перед проездом по повреждённому мосту.

Examples

standard input	standard output
120 8 55 10 35 12 18 50 41 10	5
30 4 40 10 20 10	0

Problem E. Easiest Chess Position

Input file: *standard input*
Output file: *standard output*
Time limit: 2 second
Memory limit: 512 mebibytes

Вам задано расположение белого короля и белой ладьи, а также чёрного короля на шахматной доске. Расположение фигуры задаётся двумя символами — буквой от “a” до “h”, задающей вертикаль, и цифрой от 1 до 8, задающей горизонталь. В данный момент очередь хода за чёрными.

Напомним, что ладья атакует все поля, находящиеся с ней на одной горизонтали или вертикали, кроме случаев, когда между данным полем и ладьёй в соответствующем направлении есть другая фигура. Король атакует все поля, соседние с ним по стороне или по углу и может делать ход (как со взятием занимающей его фигуры другого цвета, так и без) на любое из таких полей, которое не находится под атакой фигуры другого цвета.

Требуется классифицировать позицию в соответствии со следующими вариантами:

- странный — чёрный король находится под атакой белого,
- обычная позиция — чёрный король не находится под атакой и может сделать ход,
- шах — чёрный король находится под атакой и может сделать ход,
- пат — чёрный король не находится под атакой и не может сделать ход,
- мат — чёрный король находится под атакой и не может сделать ход.

В случае странной позиции дальнейшего анализа не производится, позиция сразу же классифицируется как странная.

Input

В первой строке заданы три пары символов. Первый символ каждой пары — буква от “a” до “h”, второй — цифра от 1 до 8. Эти три пары задают позиции белого короля, белой ладьи и чёрного короля соответственно. Соседние пары символов разделены пробелами. Гарантируется, что никакие две фигуры не занимают одну и ту же позицию.

Output

В случае, если позиция является странной, выведите “Strange”, в случае, если обычной — “Normal”, в случае, если является шахом — “Check”, если является патом — “Stalemate”, если является матом — “Checkmate”.

Examples

standard input	standard output
b2 a3 c4	Normal
a1 b2 e2	Check

Problem F. Frogs

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 512 mebibytes

Перед открытием олимпиады членов жюри повезли на экскурсию к водопаду. Но водопад был закрыт на профилактику. После этого зебра Гиппо решила осмотреть окрестности самостоятельно и добралась до знаменитого Длинного Барьерного болота.

Болото представляет собой бесконечно длинную последовательность кочек, занумерованных последовательными неотрицательными целыми числами. Для каждого $i \geq 0$ высота кочки i равна остатку от деления x^i на p .

В начальный момент времени k лягушек, занумерованных последовательными целыми числами от 1 до k , находятся на кочке 0, при этом усталость каждой лягушки равна 1. Понаблюдав за лягушками, Гиппо заметила, что лягушки двигаются в соответствии со следующими правилами:

1. Лягушка с номером 1 двигается на одну кочку вперёд, и её усталость увеличивается на величину, равную высоте новой кочки.
2. Оставшиеся лягушки двигаются по очереди, начиная со второй, так: i -я лягушка двигается на одну кочку вперёд, если $i - 1$ -я лягушка тоже двигалась и усталость $i - 1$ -й лягушки делится на m (в этом случае усталость i -й лягушки увеличивается на величину, равную высоте кочки, на которую она попала), иначе она остаётся на месте (и тогда её усталость не меняется).
3. Если расстояние между первой и k -й лягушками не менее d , лягушки прекращают движение. В противном случае процесс повторяется, начиная с пункта 1.

Вычислите, на какой кочке окажется первая лягушка в момент окончания движения.

Input

Вход содержит пять целых чисел x ($1 \leq x \leq p - 1$), p ($2 \leq p \leq 10^5$), k ($2 \leq k \leq 10$), m ($2 \leq m \leq 10$) и d ($1 \leq d \leq 10^{12}$).

Гарантируется, что число p является **простым**.

Output

Выведите номер кочки, на которой окажется первая лягушка в момент, когда лягушки прекратят движение.

Examples

standard input	standard output
1 2 3 2 10	14
58 10007 10 10 123456789012	123456789143

Problem G. General and his Army

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 512 mebibytes

Генерал Байтазар построил личный состав своей армии в одну шеренгу на плацу по росту. Рядовые байтландской армии носят фиолетовые парадные мундиры, сержанты - белые. При этом среди личного состава полка один сержант приходится ровно на k солдат.

Для распределения солдат по взводам командир использовал следующий метод: каждую минуту он выбирал k рядовых и одного сержанта, которые на начало минуты находились в одном «связном» участке строя (то есть между любыми двумя выбираемыми в одну и ту же минуту военнослужащими не должно быть пустого места в строю) и приказывал им выйти из шеренги, сформировав очередной взвод. Если военнослужащий выходит из шеренги, то его место остаётся пустым.

Требуется помочь генералу и найти такую последовательность выбора, чтобы в результате его армия была полностью поделена на взводы.

Input

Первая строка входа содержит два целых числа n и k ($2 \leq n \leq 1\,000\,000$, $1 \leq k \leq n-1$) — численность армии Байтазара и количество солдат, приходящихся на одного сержанта.

Во второй строке заданы n символов «v» и «w», задающих цвета мундиров; i -й из этих символов равен «v», если на i -м месте стоит рядовой, и «w», если там стоит сержант. Гарантируется, что n делится на $k+1$ и для каждого теста существует решающая задачу последовательность выбора.

Output

Выведите $n/(k+1)$ строк, задающих выбор командира в начале каждой минуты. i -я строка должна содержать $k+1$ номеров военнослужащих, выбираемых в соответствующую минуту. Места в шеренге пронумерованы с единицы. Принимается любая последовательность, удовлетворяющая условию задачи.

Examples

standard input	standard output
12 2	10 11 12
wwwvvvvvvvvwv	1 8 9
	2 6 7
	3 4 5

Problem H. How Many Triangles?

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 512 mebibytes

Антимонопольный комитет Байтландии принял к рассмотрению иск компании «Hexomino Inc» к конкуренту, «Triangle Limited». Суть иска заключается в том, что, используя тот факт, что геометрический объект, присутствующий в названии компании, является более простым, Triangle Limited нарушает правила рекламного бизнеса, размещая свои логотипы не только в официально оплаченных рекламных материалах.

Эксперты установили, что в качестве логотипа «Triangle Limited» устойчиво распознаётся любая конструкция на плоскости высотой h пикселей, i -й сверху ряд которой содержит $2i - 1$ пикселей, а все ряды отцентрированы так, что получившаяся фигура имеет вертикальную ось симметрии. «Повёрнутые» и «перевёрнутые» варианты данной картинки логотипами не считаются.

По заданной квадратной монохромной картинке, в которой символом '.' обозначены неактивные пиксели, а символом '*' — активные, вычислите общее количество объектов, которые распознаются в качестве логотипов «Triangle Limited».

Input

В первой строке входного файла задано число N — длина стороны квадратной картинке ($1 \leq N \leq 2000$). Последующие N строк задают картинку. Каждая из этих строк состоит только из символов '.' и '*' и содержит ровно N символов.

Output

Выведите одно число — количество обнаруженных на картинке логотипов «Triangle Limited».

Examples

standard input	standard output
5***. .***. *****	16

Problem I. Integer Sum

Input file: *стандартный ввод*
Output file: *стандартный вывод*
Time limit: 2 секунды
Memory limit: 512 мебибайт

Миша недавно узнал, что факториал числа n — это произведение всех натуральных чисел от 1 до n включительно. Факториал числа n имеет обозначение $n!$; так, например, $1! = 1$, $2! = 1 \cdot 2 = 2$, $3! = 1 \cdot 2 \cdot 3 = 6$. По определению $0! = 1$.

Мише понравилось вычислять факториалы маленьких чисел. Однако он быстро понял, что факториалы чисел, больших десяти, очень сложно вычислять вручную. Тогда он стал искать лишь несколько последних цифр, но обнаружил, что они слишком часто оказываются нулями: к примеру, в числе $25!$, если верить его подсчётам, последние шесть цифр — нули. Гораздо более интересной задачей оказалось найти последнюю ненулевую цифру $n!$, но Миша справился и с ней.

Руководитель математического кружка, в котором учится Миша, узнал о его успехах и дал ему более сложную задачу: найти последнюю ненулевую цифру суммы k -х степеней факториалов целых чисел от 0 до n , где k — неотрицательное целое число. Правда, руководитель сделал скидку на то, что Миша ещё учится в шестом классе, и поставил ограничение $k \leq 3$, $n \leq 100$. Впрочем, считать Мише всё равно придётся очень долго...

Помогите Мише с помощью компьютера решить поставленную задачу. Напишите программу, которая по заданным значениям n и k вычисляет последнюю ненулевую цифру суммы k -х степеней факториалов всех целых чисел в промежутке от 0 до n .

Input

В первой строке заданы два целых числа — положительное число n , не превосходящее 100, и неотрицательное число k , не превосходящее 3.

Output

Выведите одно число — последнюю ненулевую цифру суммы k -х степеней факториалов целых чисел от 0 до n включительно.

Examples

стандартный ввод	стандартный вывод
4 1	4
3 2	2

Problem J. Just Count Them!

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 512 mebibytes

На плоскости расположен набор из n окружностей. Каждая окружность задана тремя различными точками, через которые она проходит. Найдите наибольшее количество окружностей, имеющих один и тот же радиус.

Input

В первой строке входного файла задано целое число n ($1 \leq n \leq 1400$). В последующих n строках заданы по шесть целых чисел $x_1, y_1, x_2, y_2, x_3, y_3$ — координаты трёх точек, не лежащих на одной прямой и задающих соответствующую окружность ($0 \leq x_i, y_i \leq 1400$).

Output

Одно целое число — количество окружностей, имеющих наиболее часто встречающийся в наборе радиус.

Examples

standard input	standard output
3 0 0 0 1 1 0 0 0 2 0 0 2 1 1 1 2 2 1	2

c

Problem K. Knights

Input file: *standard input*
Output file: *standard output*
Time limit: 6 seconds
Memory limit: 512 mebibytes

На шахматной доске $W \cdot H$ находятся два коня. Один из них является (p_1, q_1) - конем (т.е. при своем ходе он движется на p_1 клеток по горизонтали и на q_1 клеток по вертикали либо на q_1 клеток по горизонтали и на p_1 клеток по вертикали; обычный шахматный конь является $(2, 1)$ -конем) и находится в клетке (x_1, y_1) , а второй является (p_2, q_2) - конем и находится в клетке (x_2, y_2) . Свои ходы кони делают одновременно. Напишите программу, которая выводит кратчайшую последовательность ходов, приводящую коней в одну и ту же клетку, либо -1, если такой последовательности ходов не существует.

Input

Десять целых положительных чисел, не превосходящих 30 - $W, H, p_1, q_1, p_2, q_2, x_1, y_1, x_2, y_2$, $1 \leq x_1, x_2 \leq W, 1 \leq y_1, y_2 \leq H$. Гарантируется, что исходные позиции коней не совпадают.

Output

В первой строке выведите требуемое минимальное число ходов K . Затем в следующих $K + 1$ строчках выведите все состояния пути, включая начало, конец и все промежуточные состояния (в порядке прохода от начального состояния к конечному).

Если искомого пути нет, выведите в единственной строке число -1.

Examples

standard input	standard output
4 4 1 2 2 1 1 1 2 2	2 1 1 2 2 3 2 3 4 1 3 1 3
4 4 1 2 2 1 1 1 3 2	-1

Problem L. Launching Fireballs

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 512 mebibytes

Алиса и Боб играют в новую компьютерную игру. Каждый игрок в этой игре управляет армией из N магов. В начальной позиции маги с каждой стороны выстроены в ряд и пронумерованы последовательными целыми числами от 1 до N .

Первым атакует первый (в соответствии с нумерацией) маг Алисы, затем — первый маг Боба, затем — второй маг Алисы, затем — второй маг Боба и так далее.

Во время атаки маг выбирает представителя противоположного войска, в направлении которого будет брошен fireball, после чего выполняет заклинание. В случае, если маг попадает в противника, тот выбывает из игры и его очередь хода переходит к следующему по очереди **вне зависимости от игрока**. Например, если первый маг Алисы успешно поразил первого мага Боба, тот не сможет сделать следующий ход, и ход переходит ко второму магу Алисы.

Для каждого мага вероятность успешно выполнить заклинание постоянна и не зависит от выбора противника.

Процесс идёт до тех пор, пока в очереди на атаку остался хотя бы один маг. Как только среди участвующих в игре магов не осталось таких, которые бы не сделали ход, игра заканчивается.

Победителем считается игрок, у которого после окончания игры осталось больше магов. Если магов осталось поровну, соревнование завершилось вничью.

По заданным вероятностям попадания для каждого из магов вычислите вероятности победы Алисы и вероятность ничьей при условии, что каждый маг выбирает цель таким образом, чтобы максимизировать шансы победы своей команды, а в случае, если таких вариантов несколько — чтобы минимизировать шансы оппонентов на победу.

Input

В первой строке ввода задано одно целое число $2 \leq N \leq 8$ — количество магов в каждой команде. Во второй строке заданы вероятности точного заклинания для управляемых Алисой магов, в порядке возрастания их номеров. В третьей аналогичным образом заданы вероятности точного заклинания для управляемых Бобом магов. Все вероятности являются неотрицательными числами, не превосходящими 1, и заданы с двумя знаками после десятичной точки.

Output

Выведите два числа — вероятности победы Алисы и вероятность ничьей с точностью не хуже 0.01.

Examples

standard input	standard output
2 0.80 1.00 1.00 0.80	0.80 0.00
2 0.33 0.33 0.33 0.33	0.41 0.27