

## Problem A. Candies

Input file: *standard input*  
Output file: *standard output*  
Time limit: 1 second  
Memory limit: 512 mebibytes

$n$  школьников играют в следующую игру. Школьники занумерованы последовательными целыми числами  $1, 2, \dots, n$ . Изначально  $i$ -й школьник держит в руке  $a_i$  конфет.

Игра проходит в  $m$  раундов. В каждом раунде школьник, у которого в настоящий момент меньше всего конфет, получает  $x$  конфет. Если наименьшее количество конфет в данном раунде сразу у нескольких школьников, то дополнительные конфеты достаются школьнику с наименьшим номером.

Школьник с номером 1 является магом и обладает специальной способностью — он может сгенерировать некоторое дополнительное количество конфет ( $y$  или менее) перед началом игры. Какое наибольшее количество конфет у него может быть после  $m$  раундов?

### Input

Первая строка входа содержит 4 целых числа  $n, m, x, y$  ( $1 \leq n, m \leq 200000, 1 \leq x, y \leq 10^9$ ).

Вторая строка содержит  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^9$ ).

### Output

Выведите одно целое число — наибольшее количество конфет, которое может оказаться у первого школьника после  $m$  раундов.

### Examples

standard input	standard output
2 1 2 2 1 2	4

## Problem B. Chessboard Game

Input file: *standard input*  
Output file: *standard output*  
Time limit: 1 second  
Memory limit: 512 mebibytes

Алиса и Боб играют в игру, полем для которой является шахматная доска из  $(n+1)$  строк и  $(m+1)$  столбцов. Строки занумерованы с нуля сверху вниз, столбцы — с нуля слева направо.

Клетки  $(0, 1), (0, 2), \dots, (0, m), (1, 0), (2, 0), \dots, (n, 0)$  являются особенными. Они могут содержать либо “чёрную дыру”, либо “белую дыру”. Игрок, попавший в “белую дыру”, немедленно выигрывает, а в “чёрную” — немедленно проигрывает, отдавая победу другому игроку.

Игра продолжается  $q$  раундов. В каждом раунде фишка ставится изначально на клетку  $(x_i, y_i)$ . Алиса и Боб ходят по очереди, Алиса ходит первой. За один ход фишка может быть сдвинута на одну клетку вверх или влево.

Для каждого раунда определите, сможет ли Алиса выиграть, если оба игрока играют оптимально.

### Input

Первая строка входа содержит 3 целых числа  $n, m, q$  ( $1 \leq n, m, q \leq 2 \cdot 10^5$ ).

Вторая строка состоит из  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $0 \leq a_i \leq 1$ ). Если клетка  $(i, 0)$  содержит “белую дыру”, то  $a_i = 0$ , в противном случае она содержит “чёрную дыру” и  $a_i = 1$ .

Третья строка состоит из  $m$  целых чисел  $b_1, b_2, \dots, b_m$  ( $0 \leq b_i \leq 1$ ). Если клетка  $(0, i)$  содержит “белую дыру”, то  $b_i = 0$ , в противном случае она содержит “чёрную дыру” и  $b_i = 1$ .

Каждая из последующих  $q$  строк содержит по два целых числа  $x_i, y_i$  ( $1 \leq x_i \leq n, 1 \leq y_i \leq m$ ).

### Output

Для каждого раунда выведите “Yes”, если Алиса может выиграть, и “No” в противном случае.

### Examples

standard input	standard output
2 2 4	No
10	Yes
11	Yes
1 1	No
1 2	
2 1	
2 2	

## Problem C. Geometric Progression

Input file: *standard input*  
Output file: *standard output*  
Time limit: 1 second  
Memory limit: 512 mebibytes

Бобу нравятся геометрические прогрессии. Он хочет узнать количество геометрических прогрессий длины 3 в последовательности  $a_1, a_2, \dots, a_n$ .

Иначе говоря, подсчитать количество троек  $(i, j, k)$  таких, что  $i < j < k$  и  $a_i \cdot a_k = a_j^2$ .

### Input

Первая строка содержит целое число  $n$  ( $1 \leq n \leq 10^6$ ).

Вторая строка содержит  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $1 \leq a_1 < a_2 < \dots < a_n \leq 10^6$ ).

### Output

Выведите одно число — искомое количество геометрических прогрессий.

### Examples

standard input	standard output
3 1 2 4	1
4 1 2 4 8	2

## Problem D. Inverse KMP

Input file: *standard input*  
Output file: *standard output*  
Time limit: 1 second  
Memory limit: 512 mebibytes

Боб только что выучил алгоритм Кнута-Мориса-Пратта.

Для строки  $S = s_1s_2\dots s_n$ ,  $\text{KMP}(S) = (f_2, f_3, \dots, f_n)$ , где  $f_i$  это максимальное  $j < i$  такое, что  $s_1s_2\dots s_j = s_{i-j+1}s_{i-j+2}\dots s_i$ .

По заданным  $f_2, f_3, \dots, f_n$  и размеру алфавита найдите количество строк  $S$  таких, что  $\text{KMP}(S) = (f_2, f_3, \dots, f_n)$ . Так как ответ может быть слишком большим, выведите его по модулю  $(10^9 + 7)$ .

### Input

Первая строка входа содержит 2 целых числа  $n$  и  $c$ , задающие длину строки и размера алфавита, соответственно ( $2 \leq n \leq 2 \cdot 10^5, 1 \leq c \leq 10^9$ ).

Вторая строка содержит  $(n - 1)$  целых чисел  $f_2, f_3, \dots, f_n$  ( $0 \leq f_i < i$ ).

Гарантируется, что количество требуемых строк непусто.

### Output

Выведите одно целое число — остаток от деления количества подходящих строк по модулю  $10^9 + 7$

### Examples

standard input	standard output
3 3 0 0	12
5 1000000000 1 2 3 4	1000000000

## Problem F. Walls

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

Однажды Гэндальф Серый оказался в лабиринте.

Лабиринт представляет собой прямоугольник из  $nm$  квадратных секций, состоящий из  $n$  строк и  $m$  столбцов. Каждая секция содержит стену, соединяющую две противоположные вершины квадрата. Таким образом, всего есть два типа секций. Гэндальф может поменять тип секции  $(i, j)$ , затратив  $c_{i,j}$  единиц магической энергии. Гэндальф хочет сделать лабиринт безопасным, то есть добиться того, чтобы в нём не было точки, целиком огороженной стенами.

Какое минимальное количество единиц магической энергии потратит Гэндальф для этого?

### Input

Первая строка содержит два целых числа  $n, m$  ( $1 \leq n, m \leq 1000$ ).

Каждая из последующих  $n$  строк содержит  $m$  символов, обозначающих направление стены в соответствующей секции.

Далее идут  $n$  строк, каждая из которых содержит по  $m$  целых чисел  $c_{i,1}, c_{i,2}, \dots, c_{i,m}$  ( $1 \leq c_{i,j} \leq 1000$ ).

### Output

Выведите одно целое число — минимальное количество единиц магической энергии, которые Гэндальф потратит на то, чтобы сделать лабиринт безопасным.

### Examples

standard input	standard output
3 3 /\/ \/ /\/ 1 3 3 3 1 3 3 3 3	2
2 2 \ \ 1000 1000 1000 1000	0

## Problem J. XOr

Input file: *standard input*  
Output file: *standard output*  
Time limit: 1 second  
Memory limit: 512 mebibytes

Задана последовательность целых чисел  $a_1, a_2, \dots, a_n$ . Разобьём её ровно на  $m$  последовательных частей. Стоимостью каждой части является побитовый *xor* всех составляющих её чисел; стоимостью разбиения является побитовый *or* всех  $m$  его частей.

Найдите минимальную стоимость разбиения.

### Input

Первая строка входа содержит два целых числа  $n, m$  ( $1 \leq n \leq 200000, 1 \leq m \leq n$ ).

Вторая строка содержит  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $0 \leq a_i \leq 10^9$ ).

### Output

Выведите одно целое число — минимальную стоимость.

### Examples

standard input	standard output
3 2 1 3 2	1
4 3 1 2 0 2	3

## Problem K. Barcode

Input file: *standard input*  
Output file: *standard output*  
Time limit: 1 seconds  
Memory limit: 512 mebibytes

Штрих-код состоит из чёрных и белых вертикальных полос, которые преобразуются при считывании в нули и единицы. Белые и чёрные полосы чередуются и могут быть толстыми или тонкими. Тонкая полоса соответствует 0, а чёрная — 1 вне зависимости от цвета.

Один из сканеров воспринимает каждую полосу на штрихкоде как столбец высотой 5 на клетчатой бумаге. Ширина тонкой полосы равна 1, ширина толстой — 2.

При считывании повреждённого штрих-кода сканер не может распознать цвет отдельных клеток. Ваша задача — по считанным данным восстановить битовую последовательность, соответствующую коду.

### Input

Первая строка входа содержит целое число  $N$  ( $1 \leq N \leq 100$ ) — ширина штрих-кода в клетках.

Каждая из следующих пяти строк содержит по  $N$  символов; символ 'X' обозначает распознанную чёрную клетку, '.' — распознанную белую, '?' — нераспознанную.

### Output

Выведите последовательность двоичных цифр, заданную на штрих-коде; если двоичная последовательность однозначно не определяется или код некорректен, выведите  $-1$ .

### Example

standard input	standard output
4 .X?? .??. ???.? ?X.? .X?.	001

## Problem L. Black Box

Input file: *standard input*  
Output file: *standard output*  
Time limit: 1 second  
Memory limit: 512 mebibytes

У Вас есть чёрный ящик, который Вы используете в качестве копилки, сбрасывая туда мелочь. Вы ходите определить, какая сумма денег собрана в данный момент в чёрном ящике, не разрушая его.

Единственный способ сделать это — взвесить ящик и попытаться определить сумму по весу.

Предположим, что вес ящика равен нулю, то есть мы можем узнать суммарный вес монет внутри. Также нам известны соответствия весов и номиналов для всех возможных монет.

Найдите, какая сумма гарантированно находится в чёрном ящике.

### Input

Первая строка входа содержит одно целое число  $W$  ( $1 \leq W \leq 10^4$ ) — суммарный вес монет в чёрном ящике. Вторая строка содержит целое число  $N$  ( $1 \leq N \leq 500$ ) — количество различных номиналов монет. Каждая из последующих  $N$  строк содержит два целых числа  $P$  и  $M$  ( $1 \leq P \leq 5 \cdot 10^4$ ,  $1 \leq M \leq 10^4$ ), где  $P$  — номинал монеты, а  $M$  — вес монеты в граммах.

### Output

Выведите одно целое число — минимальную сумму, которая может соответствовать данному весу монет. Если никакой набор монет не может обеспечить соответствующий вес, выведите  $-1$ .

### Example

standard input	standard output
100	60
2	
1 1	
30 50	

## Problem M. Game

Input file: *standard input*  
Output file: *standard output*  
Time limit: 1 second  
Memory limit: 512 mebibytes

Эта простая игра известна с незапамятных времён. Вам предлагается расположить числа  $1, 2, 3, \dots, 2n - 1, 2n$  последовательно по кругу и затем нарисовать отрезки, соединяющие пары чисел, при этом каждое число должно быть соединено ровно с одним другим, и никакие два отрезка не могут пересекаться.

Определите, сколько существует способов соединить числа в соответствии с правилами игры.

### Input

Каждая строка входа содержит целое положительное число  $n$  ( $1 \leq n \leq 100$ ). Вход завершается строкой, содержащей число  $-1$ , обрабатывать которое не следует.

### Output

Для каждого  $n$  выведите требуемое количество способов соединения.

### Example

standard input	standard output
1	1
2	2
3	5
4	14
5	42
-1	

## Problem N. Ordered Sequences

Input file: *standard input*  
Output file: *standard output*  
Time limit: 1 second  
Memory limit: 512 mebibytes

Рассмотрим строки, составленные из символов 'A' и 'B', такие, что первым символом является 'B'. Упорядочим их сначала по длине (меньшие идут раньше), затем лексикографически и занумеруем, начиная с единицы.

По заданной последовательности найдите её номер.

### Input

Первая строка входа содержит целое число  $1 \leq n \leq 1000$  — количество тестовых примеров.

Каждая из последующих  $n$  строк содержит непустую строку, состоящую из 'A' и 'B' и начинающуюся с 'B'. Длина строки не превосходит 24.

### Output

Для каждого тестового примера выведите номер соответствующей строки в соответствии с введённой в задаче нумерацией.

### Example

standard input	standard output
3	1
B	2
BA	3
BB	