Задача A. Диофантово уравнение

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	0.25 секунд
Ограничение по памяти:	256 мегабайт

Даны натуральные числа a, b и c. Решите в целых числах уравнение $ax+by=c$. Среди множества решений следует выбрать такое, где x имеет наименьшее неотрицательное значение.

Формат входных данных

Первая строка содержит три целых числа a и b и c ($1 \leqslant a, b, c \leqslant 10^9$).

Формат выходных данных

Выведите искомые x и y через пробел. Если решения не существует, выведите одну строку «`Impossible`».

Примеры

стандартный ввод	стандартный вывод
`1 2 3`	`1 1`
`10 6 8`	`2 -2`

Задача B. Армия математиков

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	0.5 секунд
Ограничение по памяти:	256 мегабайт

У вас есть n математиков. Пусть интеллектуальность i-го математика равна a_i. Для некоторого k назовём $i_1, i_2, \ldots, i_k$ сходкой математиков, если $i_1 < i_2 < i_3 < \ldots < i_k$ и $gcd(a_{i_1}, a_{i_2}, \ldots, a_{i_k}) > 1$. Эффективность этой сходки равна $k \cdot gcd(a_{i_1}, a_{i_2}, \ldots, a_{i_k})$.

Найдите сумму эффективностей всех сходок математиков. Так как это число может быть очень большим, выведите его по модулю 1000000007 ($10^9 + 7$).

Формат входных данных

Первая строка содержит целое число n ($1 \leqslant n \leqslant 200000$) — количество математиков.

Вторая строка содержит n целых чисел $a_1, a_2, \ldots, a_n$ ($1 \leqslant a_i \leqslant 1000000$) — интеллектуальности математиков.

Формат выходных данных

Выведите одно число — ответ.

Примеры

стандартный ввод	стандартный вывод
3 3 3 1	12
4 2 3 4 6	39

Замечание

В первом примере сходки — 1, 2, 1, 2, так что ответ $1 \cdot 3 + 1 \cdot 3 + 2 \cdot 3 = 12$

Задача C. Чиселки

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.5 секунд
Ограничение по памяти:	256 мегабайт

Даны два числа n и k.

Определим q_i. Изначально есть число i. Вы можете изменять его двумя способами:

1. Умножить текущее число на какое-то простое $p \leqslant n$.

2. Разделить текущее число на какое-то простое $p \leqslant n$ (если делится).

q_i — количество различных чисел, которые можно получить, если вы можете выполнить эти операции в сумме не более k раз.

Найдите $\sum\limits_{i=1}^{n} i \cdot q_i$ по модулю $10^9 + 7$.

Формат входных данных

Первая строка содержит два целых числа n, k ($1 \leqslant n, k \leqslant 10^6$).

Формат выходных данных

Выведите одно число — ответ на задачу.

Примеры

стандартный ввод	стандартный вывод
3 1	23
4 2	82

Задача D. Странная функция

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	0.5 секунд
Ограничение по памяти:	256 мегабайт

Дан массив из n целых чисел $a_1, a_2, \ldots, a_n$, определим

$$f(l, r) = \gcd(a_l, a_{l+1}, ..., a_r) \cdot \left(\left(\sum_{i=l}^{r} a_i \right) - \max(a_l, a_{l+1}, \ldots, a_r) \right).$$

Формат входных данных

Первая строка содержит целое число n $(1 \leqslant n \leqslant 50000)$.

Вторая строка содержит n целых чисел $a_1, a_2, \ldots a_n$ $(-10^6 \leqslant a_i \leqslant 10^6)$.

Формат выходных данных

Выведите $\max\limits_{1 \leqslant l \leqslant r \leqslant n} f(l, r)$.

Примеры

стандартный ввод	стандартный вывод
4 10 4 5 6	15
5 7 12 24 6 5	144

Задача E. Чиселки и странные функции

Имя входного файла: `стандартный ввод`
Имя выходного файла: `стандартный вывод`
Ограничение по времени: 0.25 секунд
Ограничение по памяти: 256 мегабайт

Есть функция $f(n)$, где $f(1) = 1$, а для $n \leqslant 2$, $f(n)$ равно количеству различных упорядоченных пар положительных целых чисел $(x,\ y)$ таких, что $x + y = n$ и $gcd(x, y) = 1$. Число $gcd(a, b)$ равно наибольшему общему делителю a и b.

Есть функция $g(n) = \sum_{d|n} f(n/d)$. Суммирование проводится по всем положительным целым числам d, делящим n.

Определим $F_k(n)$ так:

$$F_k(n) = \begin{cases} f(g(n)) & \text{для } k = 1 \\ g(F_{k-1}(n)) & \text{для } k > 1 \text{ и } k \bmod 2 = 0 \\ f(F_{k-1}(n)) & \text{для } k > 1 \text{ и } k \bmod 2 = 1 \end{cases}$$

Найдите $F_k(n)$ по модулю 100000007.

Формат входных данных

В единственной строке находятся два целых числа n ($1 \leqslant n \leqslant 10^{12}$) и k ($1 \leqslant k \leqslant 10^{12}$).

Формат выходных данных

Выведите одно целое число — значение $F_k(n)$ по модулю 100000007.

Примеры

стандартный ввод	стандартный вывод
`7 1`	`6`
`10 2`	`4`

Замечание

В первом примере есть 6 различных упорядоченных пар $(1, 6)$, $(2, 5)$, $(3, 4)$, $(4, 3)$, $(5, 2)$ и $(6, 1)$, удовлетворяющих $x + y = 7$ и $gcd(x, y) = 1$. Поэтому $f(7) = 6$. В итоге, $F_1(7) = f(g(7)) = f(f(7) + f(1)) = f(6 + 1) = f(7) = 6$.

Задача F. Кто не будет решать математику — пойдёт красить забор

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.25 секунд
Ограничение по памяти:	256 мегабайт

Миша не любит математику. Из-за этого он не смог решить сложную задачу на Всероссе, не стал призёром и не получил 150000 руб. от Москвы. Чтобы хоть как-то сводить концы с концами Мише приходится подрабатывать, а именно — красить заборы.

Мише очень нравятся зебры, поэтому он пытается найти их везде где только можно. Миша должен покрасить забор на даче и ему выдали неограниченное количество белой и чёрной краски. Забор является последовательностью досок, некоторые из которых уже покрашены в белый или чёрный цвет, а остальные ещё нет. Менять цвета уже покрашенных досок запрещается, а для остальных Миша может выбрать цвета по своему усмотрению. В данной задаче забор представляется строкой, состоящей из символов «`0`», «`1`» и «`?`», означающих белую доску, чёрную доску и ещё не окрашенную доску соответственно.

Миша считает, что забор похож на зебру, если существуют целые числа a и b ($a > 0, b \geqslant 0$), такие что первые a досок забора являются белыми, следующие b досок являются чёрными, затем снова идут a белых досок, далее опять b чёрных и так далее, при этом последний блок может быть не полным. Например, заборы, описываемые строками «`01101`» ($a = 1, b = 2$), «`000`» ($b = 0$, a может быть любым целым положительным числом) и «`00110011`» ($a = 2, b = 2$) являются зебрами, а «`01001`» и «`101010`» — нет.

Помогите Мише раскрасить оставшиеся доски таким образом, чтобы забор являлся зеброй для каких-нибудь чисел a и b ($a > 0, b \geqslant 0$). Поскольку Миша мечтает покрасить в чёрный цвет всё что он видит, то если подходящих раскрасок забора несколько, выберите среди них ту, в которой как можно больше чёрных досок. Среди таких раскрасок разрешается выбрать любую.

Формат входных данных

Входные данные содержат единственную строку s ($1 \leqslant |s| \leqslant 300000$), состоящую из символов «`0`», «`1`» и «`?`».

Формат выходных данных

Если невозможно раскрасить ещё не покрашенные доски забора таким образом, чтобы он был похож на зебру, то выведите -1 в единственной строке выходных данных. В противном случае выведите какое-нибудь решение с максимальным возможным количеством чёрных досок. Решение выводите как строку из символов «`0`» и «`1`», означающих белую и чёрную доску соответственно.

Примеры

стандартный ввод	стандартный вывод
0?	01
0110?	01101
10?	-1
011011	011011
101	-1

Задача G. k-суммы

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	0.25 секунд
Ограничение по памяти:	64 мегабайта

Неизвестный массив состоит из n целых чисел. k-сумма этого массива получается разделением его на подотрезки длины k и суммированием чисел в каждом из подотрезков. Если n не делится на k нацело, то последний подотрезок содержит меньше чем k слагаемых. Другими словами, k-сумма — массив, который можно представить как $(x[1] + \ldots + x[k])$, $(x[k+1] + \ldots + x[2k])$ и так далее, где последняя сумма, содержащая $x[n]$, может состоять из менее чем k слагаемых. Например, 5-сумма массива из 13 элементов состоит трех сумм (сумма элементов 1-5, сумма элементов 6-10 и сумма элементов 11-13). Очевидно, что нельзя однозначно восстановить изначальный массив по одной его k-сумме, но это становится возможным, если известны несколько его k-сумм для разных k.

Для заданного n и множества $k_1, k_2, \ldots, k_m$, определите сколько элементов изначального массива можно было бы восстановить, если бы были известны все слагаемые каждой k-суммы. Не составляет труда показать, что количество восстановленных элементов не зависит от самих слагаемых.

Формат входных данных

Первая строка входных данных содержит два целых числа n и m — длина изначального массива и количество k-сумм ($1 \leqslant n \leqslant 10^9$, $1 \leqslant m \leqslant 10$).

Вторая строка содержит m различных целых чисел $k_1, k_2, \ldots, k_m$ ($1 \leqslant k_i \leqslant n$).

Формат выходных данных

Выведите единственное целое число — количество элементов изначального массива, которые можно было бы однозначно восстановить, если были бы известны все слагаемые каждой k-суммы.

Примеры

стандартный ввод	стандартный вывод
3 1 2	1
6 2 2 3	2
123456789 3 5 6 9	10973937