

Технокубок 2019 - Отборочный Раунд 1

А. В поисках простой задачи

1 секунда, 256 мегабайт

Начиная очередной чемпионат, координаторы Codeforces всегда стараются сделать первую задачу максимально простой. В этот раз координатор выбрал некоторую задачу и спросил у n человек, достаточно ли она простая. Каждый человек ответил, что задача простая, или что задача сложная.

Если хоть кто-то из этих n человек ответил, что задача сложная, ее нужно заменить, иначе не нужно. По данным ответам определите, нужно ли заменить задачу.

Входные данные

В первой строке задано одно целое число n ($1 \leq n \leq 100$) — количество опрошенных человек.

Во второй строке заданы n целых чисел, каждое из которых равно либо 0, либо 1 — ответы опрошенных. Если i -е число равно 0, то i -й человек считает задачу простой, а если оно равно 1, то i -й человек считает задачу сложной.

Выходные данные

Выведите одно слово: «EASY», если задача по мнению всех опрошенных является легкой, либо «HARD», если хотя бы один опрошенный считает задачу сложной.

Вы можете выводить каждую из букв в любом регистре (как строчную или как заглавную): «EASY», «easy», «EaSY» и «eAsY» будут распознаны корректно.

Входные данные

```
3
0 0 1
```

Выходные данные

```
HARD
```

Входные данные

```
1
0
```

Выходные данные

```
EASY
```

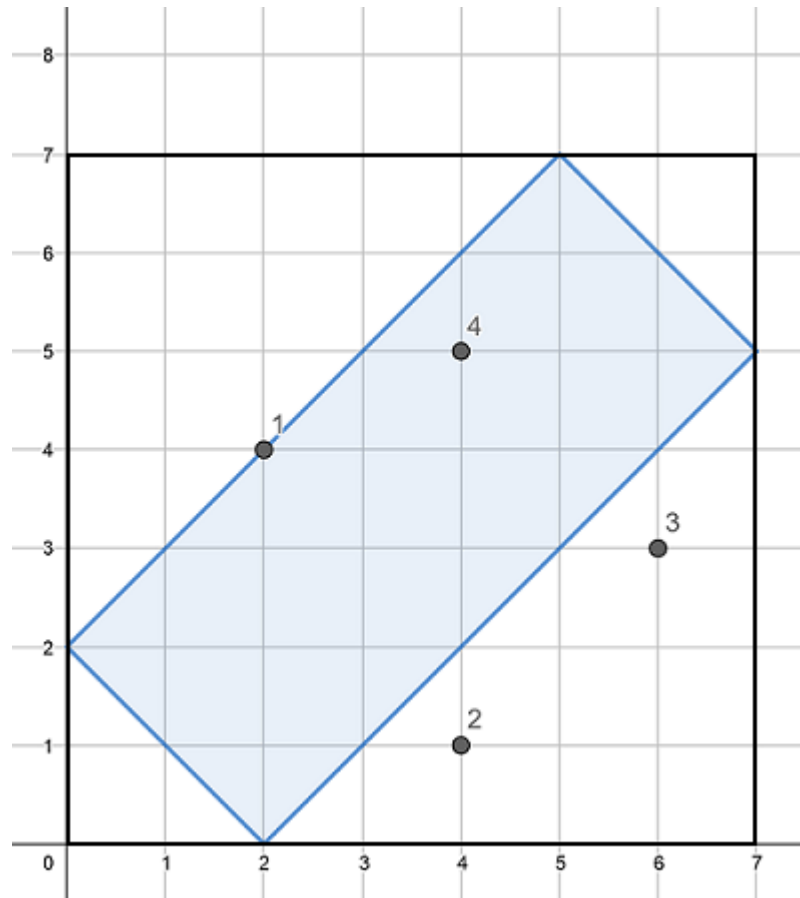
В первом примере третий человек считает, что задача слишком сложная, поэтому ее надо заменить.

Во втором примере единственный человек считает, что задача простая, поэтому ее не обязательно заменять.

В. Вася и кукурузное поле

1 секунда, 256 мегабайт

У Васи есть кукурузное поле, описываемое двумя числами — n и d . Кукурузное поле можно представить в декартовой системе координат как прямоугольник с углами в точках $(0, d)$, $(d, 0)$, $(n, n - d)$ и $(n - d, n)$.



Пример кукурузного поля при $n = 7$ и $d = 2$.

Также Вася знает координаты m кузнециков. Кузнецик номер i находится в точке (x_i, y_i) . Так как Вася не любит, когда кузнички поедают его кукурузу, то про каждого кузничка Вася хочет знать, находится ли он на поле или нет. Считается, что кузнецик находится на кукурузном поле, если он находится внутри или на границах прямоугольника, описанного выше.

Помогите Васе! Сообщите про каждого кузничка, находится ли он на кукурузном поле.

Входные данные

Первая строка содержит два целых числа n и d ($1 \leq d < n \leq 100$).

Вторая строка содержит одно целое число m ($1 \leq m \leq 100$) – количество кузничиков.

i -я из следующих m строк содержит два целых числа x_i и y_i ($0 \leq x_i, y_i \leq n$) – положение i -го кузничика.

Выходные данные

Выведите m строк. В i -й строке выведите «YES», если i -й кузничик находится на кукурузном поле. Иначе в i -й строке выведите «NO».

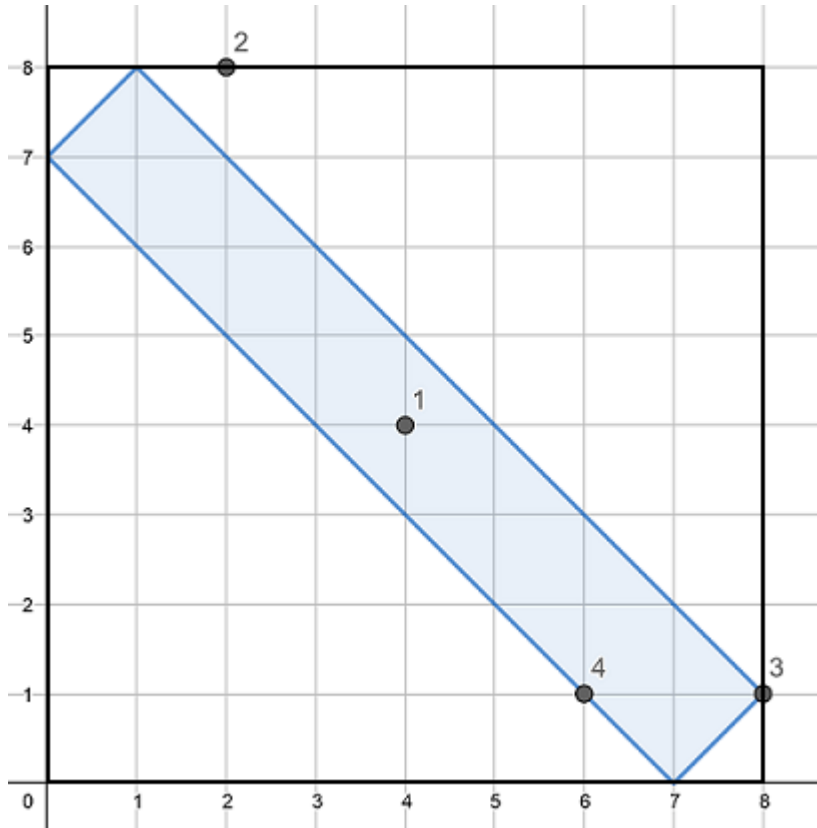
Вы можете выводить каждую из букв в любом регистре (строчную или заглавную).

ВХОДНЫЕ ДАННЫЕ	
7	2
4	
2	4
4	1
6	3
4	5
ВЫХОДНЫЕ ДАННЫЕ	
YES	
NO	
NO	
YES	

ВХОДНЫЕ ДАННЫЕ	
8	7
4	
4	4
2	8
8	1
6	1
ВЫХОДНЫЕ ДАННЫЕ	
YES	
NO	
YES	
YES	

Кукурузное поле из первого примера изображено на рисунке выше. Внутри него находятся кузнечики 1 (с координатами (2, 4)) и 4 (с координатами (4, 5)).

Кукурузное поле из второго примера изображено на рисунке ниже. Внутри него находятся кузнечики 1 (с координатами (4, 4)), 3 (с координатами (8, 1)) и 4 (с координатами (6, 1)).



С. Вася и золотой билет

1 секунда, 256 мегабайт

Недавно Вася нашел золотой билет — последовательность из n цифр $a_1 a_2 \dots a_n$. Вася считает билет счастливым, если его можно разбить на два или более непересекающихся отрезка с одинаковой суммой цифр на каждом отрезке. Например, билет 350178 счастливый, т. к. его можно разбить на три отрезка 350, 17 и 8: $3 + 5 + 0 = 1 + 7 = 8$. Заметим, что каждая цифра должна принадлежать **ровно** одному отрезку.

Помогите Васе! Сообщите ему, является ли найденный им золотой билет счастливым.

Входные данные

В первой строке содержится целое число n ($2 \leq n \leq 100$) — количество цифр в билете.

Вторая строка содержит n цифр $a_1 a_2 \dots a_n$ ($0 \leq a_i \leq 9$) — сам билет. Цифры записаны без пробелов.

Выходные данные

В единственной строке выведите «YES», если найденный Васей билет является счастливым. В противном случае выведите «NO» (в любом случае без учета регистра).

входные данные
5 73452
выходные данные
YES

входные данные
4 1248
выходные данные
NO

В первом примере билет можно разбить на 7, 34 и 52:
 $7 = 3 + 4 = 5 + 2$.

Во втором примере невозможно разбить билет на равные по сумме цифр отрезки.

D. Вася и треугольник

1 секунда, 256 мегабайт

У Васи есть три целых числа n , m и k . Он хочет найти три целочисленные точки (x_1, y_1) , (x_2, y_2) , (x_3, y_3) такие что $0 \leq x_1, x_2, x_3 \leq n$, $0 \leq y_1, y_2, y_3 \leq m$ и площадь треугольника, образованная этими точками равна $\frac{nm}{k}$.

Помогите Васе! Найдите такие три точки (если возможно). Если существует несколько решений, выведите любое из них.

Входные данные

В единственной строке заданы три целых числа n , m , k ($1 \leq n, m \leq 10^9$, $2 \leq k \leq 10^9$).

Выходные данные

Если не существует точек, удовлетворяющих описанным выше условиям, в первой строке выведите «NO».

В противном случае в первой строке выведите «YES». В следующий трех строках выведите по два целых числа x_i, y_i . Если существует несколько решений, выведите любое из них.

Вы можете выводить буквы в любом регистре (строчные или заглавные).

входные данные

4 3 3

выходные данные

YES
1 0
2 3
4 1

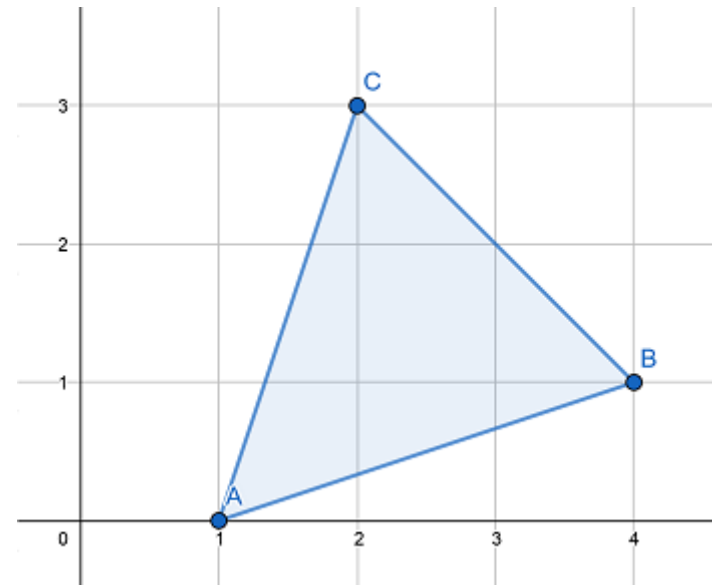
входные данные

4 4 7

выходные данные

NO

В первом примере площадь треугольника равна $\frac{nm}{k} = 4$. Сам треугольник представлен на рисунке ниже:



Во втором примере не существует ни одного треугольника с площадью $\frac{nm}{k} = \frac{16}{7}$.

E. Вася и хорошие последовательности

2 секунды, 256 мегабайт

У Васи есть последовательность a состоящая из n целых чисел a_1, a_2, \dots, a_n . За одну операцию Вася может выбрать число из последовательности и поменять местами два любых бита в его двоичной записи. Например из числа 6 (... 00000000110₂) Вася может получить числа 3 (... 0000000011₂), 12 (... 000000001100₂), 1026 (... 10000000010₂) и так далее. Вася может применять эту операцию любое количество раз (возможно, ни разу) к любому из заданных чисел.

Вася считает последовательность чисел *хорошей*, если, применяя к ней описанную выше операцию, он может получить последовательность, **побитовое исключающее ИЛИ** всех элементов которой равно 0.

Для заданной последовательности a_1, a_2, \dots, a_n Вася хочет знать количество пар целых чисел (l, r) таких, что $1 \leq l \leq r \leq n$ и последовательность a_l, a_{l+1}, \dots, a_r является хорошей.

Входные данные

Первая строка содержит одно целое число n ($1 \leq n \leq 3 \cdot 10^5$) — длину последовательности.

Вторая строка содержит n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^{18}$) — саму последовательность a .

Выходные данные

В единственной строке выведите одно целое число — количество пар целых чисел (l, r) таких, что $1 \leq l \leq r \leq n$ и последовательность a_l, a_{l+1}, \dots, a_r является хорошей.

ВХОДНЫЕ ДАННЫЕ
3 6 7 14
ВЫХОДНЫЕ ДАННЫЕ
2

ВХОДНЫЕ ДАННЫЕ
4 1 2 1 16
ВЫХОДНЫЕ ДАННЫЕ
4

В первом примере подходят пары (2, 3) и (1, 3). Для пары (2, 3) числа можно изменить следующим образом: $a_2 = 7 \rightarrow 11$, $a_3 = 14 \rightarrow 11$ и $11 \oplus 11 = 0$, где \oplus — операция побитового исключающего ИЛИ. Для пары (1, 3) числа можно изменить следующим образом: $a_1 = 6 \rightarrow 3$, $a_2 = 7 \rightarrow 13$, $a_3 = 14 \rightarrow 14$ and $3 \oplus 13 \oplus 14 = 0$.

Во втором примере подходят четыре пары: (1, 2), (2, 3), (3, 4) и (1, 4).

Ф. Сдвигаем ящики

2.5 секунд, 256 мегабайт

Задана бесконечная клеточная полоса. На ней расположены n ящиков: i -й ящик находится в клетке a_i и имеет вес w_i . Все a_i различны; также, $a_{i-1} < a_i$ для всех корректных i .

Вы решили собрать вместе некоторые ящики. «Собрать вместе» ящики с *номерами* в отрезке $[l, r]$ означает, что вам необходимо их расположить таким образом, что их *позиции* образуют некоторый отрезок $[x, x + (r - l)]$.

За один шаг вы можете сдвинуть любой ящик в соседнюю клетку, если она не занята другим ящиком (т. е. выбрать i и изменить a_i на 1, при этом все позиции должны остаться различными). На передвижение ящика i на одну клетку вы тратите энергию, равную его весу w_i . Вы можете двигать ящики в любом порядке любое количество раз.

Весы некоторых ящиков иногда изменяются, поэтому Вам необходимо обработать запросы двух видов:

- $id\ nw$ — вес w_{id} ящика id становится равным nw .
- lr — вы должны вычислить минимальную суммарную энергию, необходимую для того, чтобы собрать вместе ящики с номерами в отрезке $[l, r]$. Так как ответ может быть слишком большим, выведите остаток от деления ответа на $1000\ 000\ 007 = 10^9 + 7$. Обратите внимание, при таком запросе ящики остаются на местах, вам нужно лишь вычислить ответ.

Обратите внимание, что минимизировать надо сам ответ, а не его остаток от деления на $10^9 + 7$. Например, если у вас два возможных ответа — $2 \cdot 10^9 + 13$ и $2 \cdot 10^9 + 14$ — то необходимо выбрать первый ответ и вывести $10^9 + 6$, даже с учетом того, что второй ответ дает остаток 0.

Входные данные

Первая строка содержит два целых числа n и q ($1 \leq n, q \leq 2 \cdot 10^5$) — количество ящиков и запросов.

Вторая строка содержит n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$) — позиции ящиков. Все a_i различны, $a_{i-1} < a_i$ для всех корректных i .

Третья строка содержит n целых чисел w_1, w_2, \dots, w_n ($1 \leq w_i \leq 10^9$) — начальные веса ящиков.

Следующие q строк содержат запросы, по одному запросу в строке.

Каждый запрос описывается одной строкой, содержащей два целых числа x и y . Если $x < 0$, то это запрос первого типа, причем $id = -x, nw = y$ ($1 \leq id \leq n, 1 \leq nw \leq 10^9$). Если же $x > 0$, то это запрос второго типа, причем $l = x, a, r = y$ ($1 \leq l_j \leq r_j \leq n$). Число x не может быть равно 0.

Выходные данные

Для каждого запроса второго типа выведите ответ на него в отдельной строке. Так как ответ может быть слишком большим, выведите остаток от деления его на $1000\ 000\ 007 = 10^9 + 7$.

ВХОДНЫЕ данные

```
5 8
1 2 6 7 10
1 1 1 1 2
1 1
1 5
1 3
3 5
-3 5
-1 10
1 4
2 5
```

ВЫХОДНЫЕ данные

```
0
10
3
4
18
7
```

Рассмотрим запросы из примера:

- 1 1: ящик только один, поэтому двигать его не надо.
- 1 5: мы можем сдвинуть ящики в отрезок $[4, 8]$:
 $1 \cdot |1 - 4| + 1 \cdot |2 - 5| + 1 \cdot |6 - 6| + 1 \cdot |7 - 7| + 2 \cdot |10 - 8| = 10$
- 1 3: мы можем сдвинуть ящики в отрезок $[1, 3]$.
- 3 5: мы можем сдвинуть ящики в отрезок $[7, 9]$.
- 3 5: изменение w_3 из 1 в 5.
- 1 10: изменение w_1 из 1 в 10. Теперь веса ящиков равны $w = [10, 1, 5, 1, 2]$.
- 1 4: мы можем сдвинуть ящики в отрезок $[1, 4]$.
- 2 5: мы можем сдвинуть ящики в отрезок $[5, 8]$.

Г. Линейный конгруэнтный генератор

2 секунды, 512 мегабайт

Вам задан генератор кортежей чисел $f^{(k)} = (f_1^{(k)}, f_2^{(k)}, \dots, f_n^{(k)})$, где $f_i^{(k)} = (a_i \cdot f_i^{(k-1)} + b_i) \bmod p_i$ и $f^{(0)} = (x_1, x_2, \dots, x_n)$. Здесь $x \bmod y$ означает остаток от деления x на y . Все p_i — простые числа.

Можно заметить, что при фиксированных наборах x_i, y_i, a_i начиная с некоторого номера кортежи $f^{(k)}$ будут повторять кортежи с меньшими номерами. Найдите максимальное количество различных кортежей (из всех $f^{(k)}$ при $k \geq 0$), которые может выдать данный генератор, если x_i, a_i, b_i — целые числа в диапазоне $[0, p_i - 1]$ и могут быть выбраны произвольно. Так как ответ может быть слишком большим, выведите остаток от его деления на $10^9 + 7$.

Входные данные

В первой строке находится целое число n ($1 \leq n \leq 2 \cdot 10^5$) — количество элементов в кортеже.

Во второй строке находятся n простых целых чисел — соответствующие модули p_1, p_2, \dots, p_n ($2 \leq p_i \leq 2 \cdot 10^6$).

Выходные данные

Выведите единственное число — максимальное количество различных кортежей по модулю $10^9 + 7$.

ВХОДНЫЕ ДАННЫЕ
4 2 3 5 7
ВЫХОДНЫЕ ДАННЫЕ
210

ВХОДНЫЕ ДАННЫЕ
3 5 3 3
ВЫХОДНЫЕ ДАННЫЕ
30

В первом примере можно выбрать следующие параметры: $a = [1, 1, 1, 1]$, $b = [1, 1, 1, 1]$, $x = [0, 0, 0, 0]$, тогда $f_i^{(k)} = k \bmod p_i$.

Во втором примере можно выбрать следующие параметры: $a = [1, 1, 2]$, $b = [1, 1, 0]$, $x = [0, 0, 1]$.