

## Задача A. Grid Coloring

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

У Snuke есть табличка с  $H$  рядами и  $W$  столбцами. Snuke хочет покрасить эту табличку в цвета  $1, 2, \dots, N$ , чтобы выполнялись следующие условия:

- Для каждого  $i$  ( $1 \leq i \leq N$ ), существует ровно  $a_i$  клеток покрашенных в цвет  $i$ . Гарантируется, что  $a_1 + a_2 + \dots + a_N = H \cdot W$ .
- Для каждого  $i$  ( $1 \leq i \leq N$ ), все клетки цвета  $i$  связаны. Это значит, что из каждой клетки цвета  $i$  мы можем добраться до любой клетки цвета  $i$  последовательно перемещаясь в соседнюю по стороне клетку цвета  $i$ .

Найдите способ покрасить таблицу, чтобы условия выполнялись. Можно показать, что решение всегда существует.

### Формат входных данных

Первая строка ввода содержит 2 целых числа  $H$  и  $W$  ( $1 \leq H, W \leq 100$ )

Вторая строка содержит одно целое число  $N$  ( $1 \leq N \leq H \cdot W$ )

Третья строка содержит  $N$  чисел  $a_i$  ( $1 \leq a_i$ ) ( $\sum_{i=1}^N a_i = H \cdot W$ )

### Формат выходных данных

Выведите способ покрасить табличку, который удовлетворяет условиям.

Вывод должен содержать  $H$  строк, в каждой из которой должно быть  $W$  чисел.

### Примеры

стандартный ввод	стандартный вывод
2 2 3 2 1 1	1 1 2 3
3 5 5 1 2 3 4 5	1 4 4 4 3 2 5 4 5 3 2 5 5 5 3
1 1 1 1	1

## Задача В. Reversible Cards

Имя входного файла: стандартный ввод  
 Имя выходного файла: стандартный вывод  
 Ограничение по времени: 1.5 секунд  
 Ограничение по памяти: 256 мегабайт

У вас имеется  $N$  карт пронумерованных от 1 до  $N$ . На каждой из них с двух сторон написано по одному целому положительному числу. На карте  $i$  записаны числа  $a_i$  и  $b_i$ .

Для каждой карты вы можете выбрать, какой стороной вверх она будет лежать. Найдите максимальное возможное количество различных чисел на верхних сторонах карт.

### Формат входных данных

Первая строка содержит целое число  $N$  ( $1 \leq N \leq 2 \cdot 10^5$ ). Следующие  $N$  строк содержат пары чисел  $a_i$  и  $b_i$  ( $1 \leq a_i, b_i \leq 4 \cdot 10^5$ ).

### Формат выходных данных

Выведите одно целое число - ответ на задачу.

### Примеры

стандартный ввод	стандартный вывод
4 1 2 1 3 4 2 2 3	4
2 111 111 111 111	1
12 5 2 5 6 1 2 9 7 2 7 5 5 4 2 6 7 2 2 7 8 9 7 1 8	8

---

## Задача C. Snuke the Wizard

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Перед вами расположены  $N$  квадратов, пронумерованных от 1 до  $N$  слева направо. На каждом квадрате записана буква. На квадрате под номером  $i$  записана буква  $s_i$ . Кроме того, на каждом квадрате изначально стоит по 1 голему.

Вы кастуете  $Q$  заклинаний, чтобы передвигать големов.  $i$ -е заклинание состоит из двух символов  $t_i$  и  $d_i$ , где  $d_i$  - это  $L$  или  $R$ . Когда вы кастуете заклинание, для каждого квадрата, на котором написана буква  $t_i$ , все големы из этого квадрата передвигаются в соседнюю с нашей клетку слева, если  $d_i = L$ , или передвигаются направо, если  $d_i = R$ .

Тем не менее если голем пытается передвинуться налево от клетки 1 или передвинуться направо от клетки  $N$ , он исчезает.

Найдите количество големов, которые останутся на доске после  $Q$  заклинаний.

### Формат входных данных

Первая строка содержит 2 числа  $N$  и  $Q$  ( $1 \leq N, Q \leq 2 \cdot 10^5$ )

Вторая строка содержит строку  $s$ , которая состоит из заглавных букв английского алфавита.

Следующие  $Q$  строк содержат пары чисел  $t_i$  и  $d_i$  ( $t_i$  - заглавная буква английского алфавита,  $d_i$  -  $L$  или  $R$ )

### Формат выходных данных

Выведите одно число - ответ на задачу.

**Примеры**

стандартный ввод	стандартный вывод
3 4 ABC A L B L B R A R	2
8 3 AABCBDDBA A L B R A R	5
10 15 SNCZWRCEWB B R R R E R W R Z L S R Q L W L B R C L A L N L E R Z L S L	3

## Задача D. Лосяш и дерево

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

В одной научной статье Лосяш увидел необычное дерево из  $N$  вершин, в каждой вершине записано одно число — 0 или 1.

Изучая интересные свойства этого дерева, Лосяш задался целью найти такой путь в этом дереве, что *xor* значений в вершинах, лежащих на этом пути равен *xor* значений в вершинах, которые являются соседними к какому-либо из концов пути, но не лежат на пути.

Помогите ему найти его жизненный путь!

Обратите внимание, что не запрещается выводить путь, концы которого совпадают.

### Формат входных данных

В первой строке вводится целое число  $N$  ( $2 \leq N \leq 200\,000$ ).

В следующей строке содержатся  $N$  целых чисел  $a_1, \dots, a_N$  — значения, записанные в вершинах ( $a_i = 0$  или 1)

В каждой из следующих  $N - 1$  строк вводится два числа  $u_i, v_i$  — очередное ребро дерева ( $1 \leq u_i, v_i \leq N$ ).

### Формат выходных данных

Если подходящего пути не существует, то выведите  $-1$ , иначе первая строка должна содержать одно целое число  $M$  ( $1 \leq M \leq N$ ) — количество вершин в пути.

В следующей строке должны быть записаны  $M$  целых чисел — вершины в пути в порядке обхода ( $1 \leq v_i \leq N$ )

### Примеры

стандартный ввод	стандартный вывод
4 0 1 1 1 1 2 1 3 1 4	3 2 1 3
2 0 1 1 2	-1
4 1 0 0 1 2 1 3 2 4 2	1 2

## Задача Е. Настольный футбол

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

После турнира по настольному футболу жители 8-го дома решили развлечься какой-нибудь интеллектуальной игрой. Они собираются играть со строкой  $s$  длины  $l$  ( $3 \leq l \leq 10^6$ ). Два игрока будут поочередно делать ходы. Во время своего хода игрок может удалить из строки любую букву, кроме первой или последней, такую, что после удаления данной буквы в строке не появилось двух подряд идущих одинаковых символов. Игрок, который не может сделать свой ход, считается проигравшим. Известно, что изначально в строке не было двух подряд идущих одинаковых символов.

Определите, кто при оптимальной последовательности операций окажется победителем в данной игре.

### Формат входных данных

Первая строка содержит одно целое число  $l$  ( $3 \leq l \leq 10^6$ ) — длина строки  $s$ . Следующая строка содержит строку  $s$ , состоящую из маленьких букв латинского алфавита. Гарантируется, что в данной строке нет идущих подряд двух одинаковых символов.

### Формат выходных данных

Выведите «First» в случае, если побеждает первый игрок. В ином случае выведите «Second». Ответ можно выводить как маленькими буквами, так и большими.

### Примеры

стандартный ввод	стандартный вывод
3 aba	Second
4 abza	First
12 abcacabcbcb	First

## Задача F. Замени на сумму

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Для развития математических навыков мама Вовы придумала для него игру «Замени на сумму». Она пишет на доске две последовательности чисел не обязательно равной длины, обозначим их  $a$  и  $b$  и засекает  $k$  секунд. За одну секунду малыш успевает взять два соседних элемента в одной из последовательностей, стереть их и записать на этом месте сумму этих элементов.

Требуется проверить, можно ли не более, чем за  $k$  секунд, сделать последовательности  $a$  и  $b$  равными?

### Формат входных данных

В первой строке вводятся числа  $n, m$  — длины последовательностей ( $1 \leq n, m \leq 3 \cdot 10^5$ ).

Во второй строке вводится последовательность целых чисел  $a$  длины  $n$ , в третьей строке вводится последовательность целых чисел  $b$  длины  $m$  ( $0 \leq a[i], b[i] \leq 10^9$ ).

В четвертой строке вводится количество секунд  $k$  ( $0 \leq k \leq 10^9$ ).

### Формат выходных данных

В единственной строке выведете «YES» или «NO» в зависимости от того, можно ли приравнять последовательности  $a$  и  $b$  за не более чем  $k$  секунд.

### Примеры

стандартный ввод	стандартный вывод
6 5 2 1 2 1 4 5 2 1 3 3 6 2	NO
6 5 2 1 2 1 4 5 2 1 3 3 6 9	YES

## Задача G. Many Moves

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

На столе лежат  $N$  квадратиков выложенных в ряд. Они пронумерованы числами  $1, 2, \dots, N$  слева направо.

У вас есть две фишки, которые изначально лежат на квадратиках  $A$  и  $B$ , соответственно. Требуется ответить на  $Q$  запросов следующего вида:

- Дано число  $x_i$ , передвиньте одну из двух фишек по вашему выбору на квадратик  $x_i$ .

На передвижение фишки на 1 квадратик влево или вправо уходит одна секунда, поэтому время, которое уходит, чтобы передвинуть фишку с квадратика  $X$  на квадратик  $Y$  равняется  $|X - Y|$  секундам.

Необходимо обработать все запросы за минимальное возможное время.

Вы можете двигать фишку только во время ответа на новый запрос и вы не можете двигать две фишки одновременно. Также не разрешается менять порядок, в котором вам даны запросы. Тем не менее допускается нахождение обеих фишек на одном квадратике одновременно.

### Формат входных данных

Первая строка содержит 4 числа  $N, Q, A, B$  ( $1 \leq N, Q \leq 2 \cdot 10^5, 1 \leq A, B \leq N$ )

Следующая строка содержит  $Q$  чисел  $x_i$  ( $1 \leq x_i \leq N$ )

### Формат выходных данных

Выведите минимальное время, за которое возможно обработать все запросы.

### Примеры

стандартный ввод	стандартный вывод
8 3 1 8 3 5 1	7
9 2 1 9 5 1	4
9 2 1 9 5 9	4
11 16 8 1 1 1 5 1 11 4 5 2 5 3 3 3 5 5 6 7	21

## Задача H. BBUVBlesort!

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Филипп получил от своей мамы последовательность из  $N$  чисел в качестве подарка на его день рождения.  $i$ -й ( $1 \leq i \leq N$ ) элемент последовательности равняется  $a_i$ . Все элементы попарно различны. Он хочет отсортировать полученную последовательность. Чтобы достигнуть своей цели он может выполнять два типа операций:

- Операция 1: выбрать подотрезок длины 2 и развернуть его.
- Операция 2: выбрать подотрезок длины 3 и развернуть его.

Филиппу не нравится Операция 1. Найдите минимальное количество раз, которое ему необходимо выполнить Операцию 1, чтобы отсортировать массив.

### Формат входных данных

Первая строка содержит число  $N$  ( $1 \leq N \leq 10^5$ ). В следующих  $N$  строках содержатся числа  $A_i$  ( $0 \leq A_i \leq 10^9$ )

### Формат выходных данных

Выведите минимальное количество Операций 1 которое Филипп должен выполнить.

### Примеры

стандартный ввод	стандартный вывод
4 2 4 3 1	1
5 10 8 5 3 2	0

## Задача I. TrBBnsformBBtion

Имя входного файла: стандартный ввод  
 Имя выходного файла: стандартный вывод  
 Ограничение по времени: 1 секунда  
 Ограничение по памяти: 256 мегабайт

Рассмотрим следующие операции над строками состоящими из букв  $A$  и  $B$ :

1. Выбрать символ строки. Если это  $A$ , то заменить его на  $BB$ . Если это  $B$ , то заменить его на  $AA$ .
2. Выбрать подстроку которая равна  $AAA$  или  $BBB$ , и удалить ее из строки.

К примеру, если первая операция выполнена на  $ABA$  и выбран первый символ, то строка становится  $BBBA$ . Если вторую операцию выполнить на  $BBBAAAA$  и символах с четвертого по шестой, то строка становится  $BBBA$ .

Операции могут быть выполнены в любом количестве и в любом порядке.

Даны две строки  $S$  и  $T$ , и  $q$  запросов  $a_i, b_i, c_i, d_i$ . Для каждого запроса скажите, можно ли из  $S_{a_i} \dots S_{b_i}$ , подстроки  $S$ , получить  $T_{c_i} \dots T_{d_i}$ , подстроку  $T$ .

### Формат входных данных

Первая строка содержит строку  $S$ . ( $1 \leq |S| \leq 10^5$ ) Вторая строка содержит строку  $T$ . ( $1 \leq |T| \leq 10^5$ ) Третья строка содержит число  $q$ . ( $1 \leq q \leq 10^5$ ) Следующие  $q$  строк описывают запросы и содержат числа  $a_i, b_i, c_i, d_i$ . ( $1 \leq a_i \leq b_i \leq |S|, 1 \leq c_i \leq d_i \leq |T|$ )

### Формат выходных данных

Выведите  $q$  строк.  $i$ -я строка должна содержать в себе ответ на  $i$ -й запрос. Если из  $S_{a_i}, S_{a_i+1} \dots S_{b_i}$  мы можем получить  $T_{c_i}, T_{c_i+1} \dots T_{d_i}$ , выведите "YES". Иначе, выведите "NO".

### Примеры

стандартный ввод	стандартный вывод
BBBAAAAABA	YES
BBBBA	NO
4	YES
7 9 2 5	NO
7 9 1 4	
1 7 2 5	
1 7 2 4	
AAAAABBBBAAABBBBAAAA	YES
BBBBAABBBBAAAAABBB	YES
10	YES
2 15 2 13	YES
2 13 6 16	YES
1 13 2 20	YES
4 20 3 20	NO
1 18 9 19	NO
2 14 1 11	NO
3 20 3 15	NO
6 16 1 17	
4 18 8 20	
7 20 3 14	

## Задача J. Logs

Имя входного файла: стандартный ввод  
 Имя выходного файла: стандартный вывод  
 Ограничение по времени: 1 секунда  
 Ограничение по памяти: 256 мегабайт

У нас есть  $N$  бревен длин  $A_1, A_2, \dots, A_N$ .

Мы можем разрезать эти бревна не более  $K$  раз суммарно. Когда мы режем бревно длины  $L$  на расстоянии  $t$  до одного из его концов ( $0 < t < L$ ), бревно разделяется на два бревна с длинами  $t$  и  $L - t$ .

Найдите минимальную возможную длину самого длинного бревна после выполнения  $K$  разрезов, и выведите это число после округления в большую сторону до ближайшего целого числа.

### Формат входных данных

Первая строка содержит 2 числа  $N$  and  $K$  ( $1 \leq N \leq 2 \cdot 10^5, 0 \leq K \leq 10^9$ )

Следующая строка содержит  $N$  чисел  $A_i$  ( $1 \leq A_i \leq 10^9$ )

### Формат выходных данных

Выведите одно число - ответ на задачу.

### Примеры

стандартный ввод	стандартный вывод
2 3 7 9	4
3 0 3 4 5	5
10 10 158260522 877914575 602436426 24979445	292638192 861648772 623690081 433933447 476190629 262703497 2110

## Задача К. Lining Up

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Во время пожарной тревоги в ряд выставили  $N$  школьников и пронумеровали их от 1 до  $N$ . Школьники безответственные и поэтому они забыли, в каком порядке стояли. Единственное, что каждый школьник смог запомнить - абсолютную разность между количеством людей до него и после него. Для человека с номером  $i$  данная разность равняется  $A_i$ .

Отталкиваясь от данной информации, найдите количество возможных способов, которыми школьники могли стоять в ряд по модулю  $10^9 + 7$ .

Стоит заметить, что школьники **ОЧЕНЬ** безответственные, поэтому они могли запомнить информацию неправильно. В таком случае может существовать ни одного корректного способа, который бы соответствовал данной информации. В таком случае ответом является 0.

### Формат входных данных

Первая строка содержит число  $n$  ( $1 \leq n \leq 10^5$ ). Вторая строка содержит  $n$  чисел  $A_i$  ( $0 \leq A_i \leq n - 1$ ).

### Формат выходных данных

Выведите количество возможных способов, которыми могли стоят школьники по модулю  $10^9 + 7$ .

### Примеры

стандартный ввод	стандартный вывод
5 2 4 4 0 2	4
7 6 4 0 2 4 0 2	0
8 7 5 1 1 7 3 5 3	16

## Задача L. FT Robot

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Вы поставили робота в начало координат на двумерной сетке. Изначально робот повернут в сторону положительного направления по оси  $x$ .

Этот робот получит инструкции в виде строки  $s$ .

$s$  состоит из двух видов букв и будет выполнена в порядке слева направо.

- $F$  : Передвиньте робота на 1 в сторону его текущего направления.
- $T$  : Поверните робота на 90 градусов по часовой стрелке или против часовой стрелки.

Цель робота - оказаться в координатах  $(x, y)$  после выполнения всех инструкций. Скажите, возможно ли достигнуть данной цели.

### Формат входных данных

Первая строка ввода содержит строку  $s$  ( $1 \leq s \leq 8000$ ) Вторая строка содержит два целых числа  $(x, y)$  ( $|x|, |y| \leq |s|$ )

### Формат выходных данных

Если цель достижима, выведите "Yes"; иначе, выведите "No".

### Примеры

стандартный ввод	стандартный вывод
FTFFFTFFF 4 2	Yes
FTFFFTFFF -2 -2	Yes
FF 1 0	No
TF 1 0	No
FFTTF 0 0	Yes
TTTT 1 0	No

---

## Задача М. Выставка

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	1024 мегабайта

В ближайшее время в Москве будет проведена выставка современного искусства. Будут выставлены  $n$  картин с номерами от 1 до  $n$ .

Для всех картин определены размер и стоимость. Размер картины  $i$  ( $1 \leq i \leq n$ ) равен  $a_i$ , а стоимость равна  $b_i$ .

В главном зале планируется выставить от 1 до  $n$  картин. Однако, учитывая эстетическое чувство посетителей выставки, мы выберем работы так, чтобы разница между размерами выставленных работ не была слишком большой. С другой стороны, мы хотим выставить как можно больше произведений с большой ценностью.

Было решено подобрать работы по следующему правилу:

- Среди отобранных для выставки работ пусть  $A_{max}$  будет наибольшим размером отобранных работ, а  $A_{min}$  будет наименьшим размером выбранных работ. Пусть  $S$  - общая стоимость выбранных произведений искусства.
- Тогда мы хотим максимизировать  $S - (A_{max} - A_{min})$ .

### Формат входных данных

- Первая строка содержит целое число  $N$  ( $2 \leq N \leq 500\,000$ ), количество картин.
- $i$ -я строка ( $1 \leq i \leq N$ ) из следующих  $N$  строк содержит два целых числа  $A_i, B_i$ , разделенных пробелом. Это означает, что размер произведения искусства  $i$  равен  $A_i$  ( $1 \leq A_i \leq 10^{15}$ ), а стоимость произведения искусства  $i$  равна  $B_i$  ( $1 \leq B_i \leq 10^9$ ).

### Формат выходных данных

Вывод должен содержать максимальное значение  $S - (A_{max} - A_{min})$ .

**Примеры**

стандартный ввод	стандартный вывод
15 1543361732 260774320 2089759661 257198921 1555665663 389548466 4133306295 296394520 2596448427 301103944 1701413087 274491541 2347488426 912791996 2133012079 444074242 2659886224 656957044 1345396764 259870638 2671164286 233246973 2791812672 585862344 2996614635 91065315 971304780 488995617 1523452673 988137562	4232545716
3 2 3 11 2 4 5	6
6 4 1 1 5 10 3 9 1 4 2 5 3	7

## Задача N. 1D Reversi

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Две лисы Jiro и Saburo играют в игру под названием 1D Reversi. В нее играют на доске с использованием черных и белых камней. Камни выкладываются в ряд и каждый игрок поочередно кладет камень в начало или конец ряда. Аналогично оригинальной версии игры Reversi, когда кладется белый камень, все черные камни между новым белым камнем и ближайшим к нему другим белым камнем (если он существует) меняют свой цвет на белый. Аналогично с черными камнями.

В середине игры что-то произошло и Saburo пришлось покинуть игру. Вам дана строка которая описывает выложенные на столе камни. Их количество  $|S|$  и каждый символ в строке  $S$  описывает цвет  $i$ -th ( $1 \leq i \leq |S|$ ) камня слева. Если  $S_i = W$ , то камень белый. Если  $S_i = B$ , то камень белый. Jiro хочет, чтобы все камни стали одинакового цвета. Назовите минимальное количество новых камней, которое ему необходимо выложить на доску, чтобы достигнуть своей цели.

### Формат входных данных

Единственная строка ввода содержит  $S$ . ( $1 \leq |S| \leq 10^5$ )

### Формат выходных данных

Выведите минимальное количество новых камней, которое Jiro необходимо выложить на доску.

### Примеры

стандартный ввод	стандартный вывод
BBBWW	1
WWWWW	0
WBWBWBWB	9

## Задача O. Manga Market

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

У Takahashi есть  $N$  любимых магазинов, которые он называет Магазин 1, Магазин 2, ..., Магазин  $N$ . Takahashi, который находится у себя дома в момент времени 0, хочет посетить некоторые из этих магазинов. У него уходит одна единица времени, чтобы добраться из дома до какого-либо из этих магазинов, либо чтобы переместиться из одного магазина в другой.

Если Takahashi попадает в Магазин  $i$  в момент времени  $t$ , он может заняться покупками только после того, как простоит в очереди в течение  $a_i \cdot t + b_i$  единиц времени. (Будем считать, что он не тратит время, когда занимается покупками.)

Все магазины закрываются в момент времени  $T + 0.5$ . Если он в этот момент времени стоит в очереди, то он не сможет попасть в магазин за покупками.

Takahashi не закупается в одном и том же магазине более 1 раза.

Найдите максимальное количество магазинов, в которых он может сделать покупки до момента времени  $T + 0.5$ .

### Формат входных данных

Первая строка содержит 2 целых числа  $N$  и  $T$  ( $1 \leq N \leq 2 \cdot 10^5, 0 \leq T \leq 10^9$ )

Следующие  $N$  строк содержат пары чисел  $a_i$  и  $b_i$  ( $0 \leq a_i, b_i \leq 10^9$ )

### Формат выходных данных

Выведите одно целое число - ответ на задачу.

### Примеры

стандартный ввод	стандартный вывод
3 7 2 0 3 2 0 3	2
1 3 0 3	0
5 21600 2 14 3 22 1 3 1 10 1 9	5
7 57 0 25 3 10 2 4 5 15 3 22 2 14 1 15	3

## Задача P. Removing coins

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Takahashi и Aoki играют в игру на дереве. Дерево содержит  $N$  вершин пронумерованных от 1 до  $N$ , и  $i$ -е из  $N - 1$  ребер соединяет вершины  $a_i$  и  $b_i$ .

В начале игры на каждой вершине лежит по монете. Начиная с Takahashi, он и Aoki будут поочередно выполнять следующую операцию:

- Выберите вершину  $v$ , которая содержит хотя бы одну монету, и удалите с нее все монеты. Затем одновременно совершите данную операцию с каждой оставшейся на дереве монетой. Пусть вершина  $u$  - вершина, на которой лежит наша монета. Пусть вершина  $w$  - сосед вершины  $u$ . Передвиньте монету в такую вершину  $w$ , что расстояние от  $v$  до  $w$  минимально.

Игрок, который не может сделать хода проигрывает. Определите, кто выиграет, если оба игрока действуют оптимально.

### Формат входных данных

Первая строка содержит число  $N$ . ( $1 \leq N \leq 2 \cdot 10^5$ )

Следующие  $N - 1$  строк содержат пары чисел  $a_i$  и  $b_i$ . ( $1 \leq a_i, b_i \leq N$ )

Гарантируется, что данный граф - дерево.

### Формат выходных данных

Выведите "First" если Takahashi выиграет, и выведите "Second" если Aoki выиграет.

### Примеры

стандартный ввод	стандартный вывод
3 1 2 2 3	First
6 1 2 2 3 2 4 4 6 5 6	Second
7 1 7 7 4 3 4 7 5 6 3 2 1	First

## Задача Q. Black and White Tree

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Вам дано дерево из  $N$  вершин, пронумерованных от 1 до  $N$ .  $i$ -е из  $N-1$  ребер соединяет вершины  $a_i$  и  $b_i$ . Изначально, каждая вершина не покрашена.

Takahashi и Aoki играют в игру, в которой они красят вершины. Первым ходит Takahashi. В этой игре они последовательно выполняют данную операцию:

- Выберите вершину, которая еще не покрашена.
- Если это ход Takahashi, то покрасьте вершину в белый цвет, если же это ход Aoki, то покрасьте ее в черный.

После того, как все вершины становятся покрашенными происходит данная процедура:

- Перекрасьте каждую белую вершину соседнюю с черной в черный цвет.

Заметьте, что все вершины перекрашиваются одновременно.

Если остается хотя бы одна белая вершина, то побеждает Takahashi. Если же все оставшиеся вершины черного цвета, то побеждает Aoki. Назовите победителя игры, если Aoki и Takahashi играют оптимально.

### Формат входных данных

Первая строка содержит число  $N$  ( $2 \leq N \leq 10^5$ )

Следующие  $N-1$  строк содержат пары чисел  $a_i$  и  $b_i$  ( $1 \leq a_i, b_i \leq n$ )

Гарантируется, что данный граф - дерево.

### Формат выходных данных

Выведите "First" если Takahashi побеждает; выведите "Second" если Aoki побеждает.

### Примеры

стандартный ввод	стандартный вывод
3 1 2 2 3	First
4 1 2 2 3 2 4	First
6 1 2 2 3 3 4 2 5 5 6	Second

## Задача R. Сумма весов

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Вам дана последовательность целых чисел  $a_1, a_2, \dots, a_n$  длины  $n$ . Весом подпоследовательности этого массива называется количество различных чисел в ней. Найдите сумму весов всех подпоследовательностей массива  $a$  по модулю  $10^9 + 7$ .

### Формат входных данных

В первой строке дано одно число  $n$  ( $1 \leq n \leq 10^5$ ) — длина последовательности  $a$ .

Во второй строке дано  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^9$ ) — последовательность  $a$ .

### Формат выходных данных

Для каждого набора входных данных выведите одно число: сумму весов всех подпоследовательностей  $a$  по модулю  $10^9 + 7$ .

### Примеры

стандартный ввод	стандартный вывод
3 1 2 3	12
5 1 2 1 2 1	52

### Замечание

Подпоследовательности из первого примера:  $[1]$ ,  $[2]$ ,  $[3]$ ,  $[1, 2]$ ,  $[1, 3]$ ,  $[2, 3]$ ,  $[1, 2, 3]$ . Во всех подпоследовательностях числа различны.

## Задача S. Красно-синие фишки

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

На поле  $3 \times 3$  расставлены 4 красных и 4 синих фишки. Центральная клетка свободна, а в остальных клетках стоит по одной фишке. Требуется переставить фишки таким образом, чтобы центральная клетка по-прежнему оставалась свободной, а цвета фишек чередовались по кругу: красный — синий — красный — синий — красный — синий — красный — синий, начиная с некоторой клетки. За один ход разрешается переставить любую фишку в соседнюю по стороне свободную клетку.

Клетки поля пронумерованы следующим образом:

0	1	2
7	8	3
6	5	4

### Формат входных данных

В единственной строке входного файла вводится строка, состоящая из 8 букв *R* (красная) или *B* (синяя) без пробелов — цвета фишек, стоящих на полях 0, 1, ..., 7 соответственно.

### Формат выходных данных

В первой строке выходного файла выведите одно целое число  $n$  — количество ходов, которое вы хотите сделать. В следующих  $n$  строках выведите по одному числу — номер клетки, из которой делается очередной ход.

Если решений несколько, выведите любое из них.

### Примеры

стандартный ввод	стандартный вывод
BRRBRRBR	4 1 2 3 8
RBRBRBRB	0

## Задача Т. Радиолокация

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Вводится новый авиарейс между городами  $A$  и  $B$ . Самолет будет лететь строго по прямой, соединяющей эти города. На протяжении всего пути необходимо, чтобы за самолетом могли наблюдать с земли при помощи радаров. Однако, радиус действия радаров ограничен, и равен  $R$ . Хуже того, радарные станции имеются только в определенных точках (например, других населенных пунктах). Руководство авиакомпании желает выяснить: можно ли следить за самолетом на протяжении всего пути из  $A$  в  $B$ . Если это возможно, то каким минимальным количеством уже имеющихся радаров можно для этого обойтись. Самолет, находящийся на границе действия радара является видимым, то есть если расстояние от радара до самолета точно равно  $R$ , то считается что радар еще действует.

### Формат входных данных

Сначала вводятся координаты точек  $A$  и  $B$  (целые числа, по модулю не превышают 1000), затем вводится число  $N$  (натуральное, не превышает 20) - количество уже построенных радаров, затем следует число  $R$  - радиус действия радара (натуральное, не превышает 1000). Затем вводится  $N$  пар чисел - координаты уже построенных радаров (целые, по модулю не превышают 1000). Наличие радаров в городах  $A$  и  $B$  возможно, но необязательно.

### Формат выходных данных

Выведите минимальное количество радаров, которое надо задействовать для сопровождения самолета на всем пути. Если сопровождение имеющимися средствами невозможно - выведите число -1.

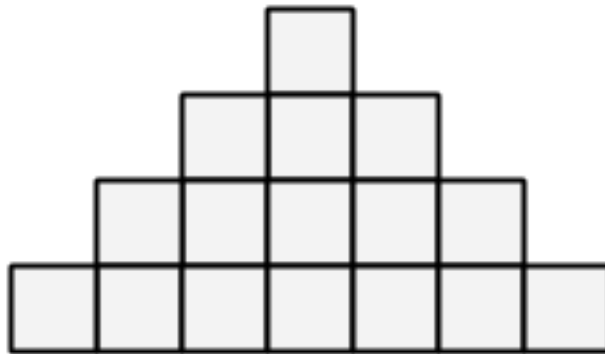
### Примеры

стандартный ввод	стандартный вывод
0 0 1 1 1 1 5 5	-1
0 0 10 0 3 5 0 0 5 0 10 0	1

## Задача U. Median Pyramid Hard

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

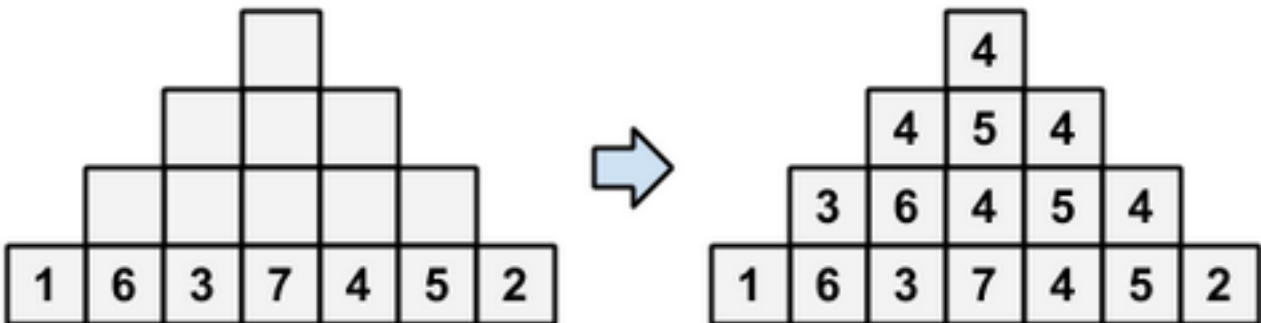
У Snuke есть пирамида с  $N$  слоями, построенная из кубиков. Слои пронумерованы от 1 до  $N$ . Для каждого  $1 \leq i \leq N$ , слой  $i$  состоит из  $2i - 1$  кубиков. Пирамида построена таким образом, что центральные кубики каждого слоя лежат друг под другом.



Пирамида с  $N = 4$  слоями.

Snuke выписал перестановку чисел  $(1, 2, \dots, 2N - 1)$  в основание пирамиды. Затем, он написал целые числа во все оставшиеся блоки, следуя следующему правилу:

- Число записанное в кубик  $b$  должно быть равно медиане трех блоков, лежащих под ним.



После этого он выкинул всю пирамиду в помойку. Единственное, что он запомнил - какие кубики лежали в основании пирамиды -  $(a_1, a_2, \dots, a_{2N-1})$ .

Найдите, какое число было записано в кубике на вершине пирамиды.

### Формат входных данных

Первая строка содержит целое число  $N$  ( $2 \leq N \leq 2 \cdot 10^5$ )

Вторая строка содержит  $2N - 1$  чисел  $a_i$

$(a_1, a_2, \dots, a_N)$  является перестановкой  $(1, 2, \dots, 2N - 1)$

### Формат выходных данных

Напишите число, которое было записано на вершине пирамиды.

**Примеры**

стандартный ввод	стандартный вывод
4 1 6 3 7 4 5 2	4
2 1 2 3	2