

## Задача А. Блочная сортировка

Имя входного файла:	стандартный ввод или <code>input.txt</code>
Имя выходного файла:	стандартный вывод или <code>output.txt</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Васе недавно подарили перестановку из  $n$  чисел. Он очень любит играть с перестановками. Играет он так: сначала перемешивает её, затем разбивает её на  $k$  непустых блоков так, что каждый элемент перестановки принадлежит ровно одному блоку. После этого Вася сортирует каждый из  $k$  блоков перестановки по отдельности. Вася хочет, чтобы после этого вся перестановка была отсортирована.

В этот раз ему подарили слишком большую перестановку, поэтому после того, как Вася перемешал перестановку, он понял, что не знает какие именно выбрать блоки. Помогите ему разбить перестановку ровно на  $k$  блоков или скажите, что это невозможно сделать.

Более формально, перестановку требуется разбить на ровно на  $k$  непустых подотрезков (каждый элемент должен принадлежать **ровно** одному подотрезку) так, чтобы если отсортировать элементы каждого из этих подотрезков по-отдельности, то в итоге получится отсортированный массив.

Перестановкой является массив, состоящий из  $n$  различных целых чисел от 1 до  $n$  в произвольном порядке. Например,  $[2, 3, 1, 5, 4]$  — перестановка, но  $[1, 2, 2]$  не перестановка (2 встречается в массиве дважды) и  $[1, 3, 4]$  тоже не перестановка ( $n = 3$ , но в массиве встречается 4).

Последовательность  $a$  является непустым подотрезком  $b$ , если  $a$  содержит хотя бы один элемент и  $a$  может быть получена из  $b$  удалением нескольких (возможно, ни одного) элементов из начала и нескольких (возможно, ни одного) элементов из конца  $b$ .

### Формат входных данных

Первая строка содержит два целых числа  $n$  и  $k$  ( $1 \leq k \leq n \leq 2 \cdot 10^5$ ) — размер перестановки и число блоков, на которые её надо разбить.

Вторая строка содержит  $n$  различных целых чисел  $p_1, p_2, \dots, p_n$  ( $1 \leq p_i \leq n$ ) — значения чисел в перестановке.

### Формат выходных данных

В единственной строке выведите  $k$  чисел — длины блоков, на которые надо разбить перестановку (сумма длин блоков должна равняться  $n$ ). Если решений несколько, разрешается вывести любое из них. Если так разбить массив на  $k$  блоков невозможно, выведите единственное число  $-1$ .

### Примеры

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

### Замечание

В первом наборе входных данных можно разбить перестановку на 2 блока так:  $[2, 3, 1]$  и  $[5, 4]$ . Отдельно их отсортировав получим:  $[1, 2, 3]$  и  $[4, 5]$ . Если соединить их, то получится отсортированная последовательность.

Во втором наборе входных данных нужно разбить перестановку на 1 блок, и это можно сделать единственным способом:  $[2, 1, 4, 3, 5]$ . Отсортируем единственный блок и получим отсортированную последовательность.

В третьем наборе входных данных можно показать, что разбиения на блоки не существует.

## Система оценки

Тесты к этой задаче состоят из 4 групп, не считая тесты из условия. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов некоторых из предыдущих групп.

Группа	Баллы	Доп. ограничения	Необх. группы	Комментарий
		$n$		
0	0	–	–	Тесты из условия.
1	21	$n \leq 100$	0	
2	22	$n \leq 1000$	0, 1	
3	18	$n \leq 10\,000$	0 – 2	
4	39	$n \leq 200\,000$	0 – 3	

## Задача В. Особенности числа

Имя входного файла: стандартный ввод или `input.txt`  
Имя выходного файла: стандартный вывод или `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 512 мегабайт

Число  $a$  называется палиндромом, если оно читается одинаково справа налево и слева направо. Число  $a$  называется особенным, если оно палиндром и число  $11 \cdot a$  тоже палиндром. Вам дано  $n$  запросов, в каждом из них вы должны найти количество особенных чисел от 1 до  $a_i$ . Так как ответ может быть большим, найдите результат по модулю  $10^9 + 7$ .

### Формат входных данных

В первой строке вводится единственное целое число  $n$  ( $1 \leq n \leq 1000$ ) — количество запросов.

В следующих  $n$  строках вводится по одному числу  $a_i$  ( $1 \leq a_i \leq 10^{5000}$ ) —  $i$ -й запрос.

### Формат выходных данных

Для каждого запроса в отдельной строке выведите единственное целое число — количество особенных чисел от 1 до  $a_i$  по модулю  $10^9 + 7$ .

### Пример

стандартный ввод	стандартный вывод
5	9
10	13
45	13
79	15
113	43
467	

### Замечание

В примере при  $a_i = 10$  особенные числа меньше 10 это: 1, 2, 3, 4, 5, 6, 7, 8, 9.

При  $a_i = 45$  особенные числа это: 1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 22, 33, 44.

### Система оценки

Тесты к этой задаче состоят из 6 групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов необходимых групп. **Offline-проверка** означает, что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.

Группа	Баллы	Доп. ограничения		Необх. группы	Комментарий
		$a_i$	$n$		
0	0	–	–	–	Тесты из условия.
1	15	$a_i \leq 70\,000$	$n \leq 100$	0	
2	9	$a_i \leq 2 \cdot 10^6$	–	0, 1	
3	14	$a_i \leq 10^8$	$n \leq 100$	0, 1	
4	11	$a_i \leq 10^{11}$	–	0–3	
5	25	$a_i \leq 10^{18}$	–	0–4	
6	26	–	–	0–5	<b>Offline-проверка</b>

## Задача С. Доставка еды

Имя входного файла:	стандартный ввод или <code>input.txt</code>
Имя выходного файла:	стандартный вывод или <code>output.txt</code>
Ограничение по времени:	3 секунды
Ограничение по памяти:	1024 мегабайта

Столица Берляндии — огромный город, в котором есть  $n$  перекрёстков, пронумерованных целыми числами от 1 до  $n$ .

Движение по городу организовано особым образом. Всего в городе есть  $m$  односторонних дорог,  $i$ -я из которых выходит из перекрёстка  $a_i$  и входит в перекрёсток  $b_i$ . У некоторых дорог есть их продолжения. При въезде на перекрёсток по дороге с номером  $i$  и выезде по дороге  $j$ , если  $j$ -я дорога является продолжением  $i$ -й, то время проезда по дороге  $j$  будет на секунду меньше времени проезда по дороге  $i$  (но если по дороге  $i$  время движения было равно 0, то по дороге  $j$  время движения тоже будет равно 0). Если же дорога  $j$  не является продолжением  $i$ -й, то машине придётся сбросить скорость для поворота и время проезда по дороге  $j$  будет равно  $c_j$ .

Более формально, для каждой дороги зафиксировано число  $d_i$ , обозначающее продолжение дороги. Если  $d_i$  равно  $-1$ , то у  $i$ -й дороги нет продолжения, а если  $d_i > 0$ , то продолжением дороги  $i$  является дорога с номером  $d_i$ .

Для каждой дороги зафиксировано время первоначального проезда по ней, равное  $c_i$ . При движении по некоторому пути время проезда по дороге с номером  $i$  определяется следующим образом:

- Если дорога  $i$  является первой на пути или не является продолжением предыдущей на пути, то время проезда по ней равно  $c_i$ .
- Если дорога  $i$  является продолжением предыдущей на пути и по предыдущей дороге машина двигалась  $x$  секунд, то время движения по текущей дороге равно  $\max(0, x - 1)$  секунде.

Недавно вы открыли новый ресторан на перекрёстке с номером 1 и хотите начать доставлять еду в разные точки города. Для каждого перекрёстка вы хотите узнать, за какое минимальное время можно доставить еду на этот перекрёсток, начиная движение с перекрёстка номер 1.

### Формат входных данных

В первой строке даны три целых числа  $n$ ,  $m$  и  $g$  ( $1 \leq n, m \leq 500\,000$ ,  $0 \leq g \leq 10$ ) — число перекрёстков в городе, число дорог в городе и номер группы тестов.

В следующих  $m$  строках даны по четыре целых числа  $a_i$ ,  $b_i$ ,  $c_i$  и  $d_i$  ( $1 \leq a_i, b_i \leq n$ ,  $1 \leq c_i \leq 10^9$ ,  $d_i = -1$  или  $1 \leq d_i \leq m$ ) — начало  $i$ -й дороги, конец  $i$ -й дороги, время первоначального проезда по  $i$ -й дороге и номер продолжения  $i$ -й дороги ( $d_i = -1$  если у  $i$ -й дороги нет продолжения).

Гарантируется, что если у дороги есть продолжение, то оно выходит из перекрёстка  $b_i$ . Также гарантируется, что если  $d_i \neq -1$ , то  $c_{d_i} \geq c_i - 1$ . Обратите внимание, что между одной и той же парой перекрёстков может проходить несколько дорог, одна дорога может быть продолжением нескольких дорог, а так же у разных дорог, входящих в перекрёсток, могут быть разные продолжения.

### Формат выходных данных

В единственной строке выведите  $n$  чисел,  $i$ -е из них должно быть равно минимальному времени, за которое можно доставить еду до перекрёстка номер  $i$ . Если доставить еду до перекрёстка номер  $i$  нельзя, выведите  $-1$ .

## Примеры

стандартный ввод	стандартный вывод
3 2 0 1 2 5 2 2 3 10 -1	0 5 9
5 4 0 1 2 5 4 3 4 10 -1 1 3 8 2 2 3 7 2	0 5 8 12 -1
4 4 0 1 2 10 3 2 2 4 3 2 4 9 4 4 1 10 1	0 10 -1 17
4 5 0 1 2 10 -1 1 3 1 3 3 4 7 4 4 2 6 5 2 2 5 5	0 1 1 1

## Замечание

В первом примере до перекрёстка 2 можно доехать по дороге 1 за 5 секунд. Чтобы доехать до перекрёстка 3, надо сначала проехать по дороге 1, а затем по её продолжению дороге 2. За счёт того, что дорога 2 является продолжением дороги 1, время движения по ней составит 4 секунды, поэтому до перекрёстка 3 можно доехать за 9 секунд.

Во втором примере можно добраться до перекрёстка 2 за 5 секунд по дороге 1. До перекрёстка 3 можно добраться за 8 секунд по дороге 3. До перекрёстка 4 можно добраться за 12 секунд по дорогам с номерами 1, 4, 2. Время движения по ним составит  $5 + 4 + 3 = 12$  секунд. До перекрёстка 5 доехать никак нельзя, так как в него не входит ни одна дорога.

В третьем примере оптимальный путь до перекрёстка 4 пройдёт по дорогам 1, 2, 3, время движения будет равно  $10 + 4 + 3 = 17$ .

## Система оценки

Тесты к этой задаче состоят из 10 групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов некоторых из предыдущих групп. Обратите внимание, прохождение тестов из условия не требуется для некоторых групп. **Offline-проверка** означает, что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.

Длинный тур отборочного этапа Открытой олимпиады школьников 2022–2023 учебного года  
Россия. 21 ноября 2022 – 15 января 2023

Группа	Баллы	Доп. ограничения			Необх. группы	Комментарий
		$n$	$m$	$c_i$		
0	0	–	–	–	–	Тесты из условия.
1	10	$n \leq 1000$	$m \leq 1000$	–	0	
2	8	$n \leq 10\,000$	$m \leq 10\,000$	–	0, 1	
3	9	–	–	–	–	У всех дорог $d_i = -1$
4	9	–	–	$c_i = 1$	–	
5	11	$n \leq 100\,000$	$m \leq 100\,000$	$c_i \leq 10$	0	
6	16	–	–	–	3	Каждая дорога является продолжением не более одной другой
7	19	$n \leq 100\,000$	$m \leq 100\,000$	–	0 – 2, 5	
8	6	$n \leq 250\,000$	$m \leq 250\,000$	–	0 – 2, 5, 7	
9	6	$n \leq 400\,000$	$m \leq 400\,000$	–	0 – 2, 5, 7, 8	<b>Offline-проверка</b>
10	6	–	–	–	0 – 10	<b>Offline-проверка</b>

## Задача D. Зима в городе К

Имя входного файла:	стандартный ввод или <code>input.txt</code>
Имя выходного файла:	стандартный вывод или <code>output.txt</code>
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Город К расположен на севере очень большой страны. И вот наконец-то в этот город пришла зима, которая продлится  $m$  дней. В городе К живут самые обычные люди большой страны, поэтому они очень не любят, когда их дом покрыт снегом.

Всего в этом городе  $n$  домов, каждый из которых изначально не покрыт снегом. Вы знаете, что утром  $i$ -го зимнего дня произойдет одно из следующих событий:

- Выпадет снег, и дома с номерами от  $l_i$  до  $r_i$  включительно покроются снегом.
- Коммунальные службы очистят от снега дома с номерами от  $l_i$  до  $r_i$  включительно.

Начиная с этого момента, Вы — мэр города К. Первое ваше задание на посту мэра — спрогнозировать уровень счастья каждого жителя. Для этого вам нужно узнать, сколько дней во время зимы каждый дом не будет покрыт снегом.

### Формат входных данных

Первая строка содержит два числа  $n$  и  $m$  ( $1 \leq n, m \leq 200\,000$ ) — количество домов в городе К, а также длительность зимы.

В следующих  $m$  строках описываются события, которые происходили в каждый из дней зимы. В  $i$ -й строке содержится символ  $c_i$  и два целых числа  $l_i$  и  $r_i$  ( $c_i = +$  или  $c_i = -$ ,  $1 \leq l_i \leq r_i \leq n$ ). Если  $c_i = +$ , то в  $i$ -й день выпал снег на отрезке домов с  $l_i$ -го по  $r_i$ -й. Если  $c_i = -$ , то в  $i$ -й день был убран весь снег с домов с  $l_i$ -го по  $r_i$ -й.

### Формат выходных данных

Для каждого дома выведите количество дней, которые он не будет покрыт снегом.

### Примеры

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

### Замечание

Рассмотрим первый пример.

В первый день зимы под снегом не будет дома 4.

Во второй день зимы под снегом не будут домов 1, 2, 4.

В третий день зимы под снегом не будет дома 1.

Поэтому ответ 2 1 0 2.

### Система оценки

Тесты к этой задаче состоят из 5 групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов некоторых из предыдущих групп. Обратите внимание, прохождение тестов из условия не требуется для некоторых групп.

Группа	Баллы	Доп. ограничения		Необх. группы	Комментарий
		$n$	$m$		
0	0	–	–	–	Тесты из условия.
1	14	$n \leq 1000$	$m \leq 1000$	0	
2	19	–	–	–	$l_i = r_i$
3	22	–	–	–	$l_i = 1$
4	18	–	–	–	Если $c_i = +$ , $c_j = -$ , то $i < j$ .
5	27	–	–	0–4	



## Задача Е. Грибные пары

Имя входного файла:	стандартный ввод или <code>input.txt</code>
Имя выходного файла:	стандартный вывод или <code>output.txt</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Недавно столичный Центр Помощи Мигрантам открыл свою школу. Школа эта очень странная — занятия идут не по 45 минут, ученики пропускают уроки, а ведет их учитель по прозвищу ГРИБ. В один момент это надоело всем (кроме учеников, конечно же), из-за чего учитель решил проучить всех прогульщиков.

ГРИБ дал ученикам следующую задачу: Дан массив  $a$ , состоящий из  $n$  целых чисел. К этому массиву приходят  $m$  запросов, состоящих из двух чисел  $x_i$  и  $y_i$ . Для каждого запроса требуется максимизировать произведение числа вхождений  $x_i$  до некоторой позиции на количество вхождений  $y_i$  начиная с этой позиции. Более формально, можно ввести следующие три функции:

- $lcnt(j, x)$  — количество вхождений числа  $x$  на префиксе массива  $a$  до позиции  $j$  включительно.
- $rcnt(j, x)$  — количество вхождений числа  $x$  на суффиксе массива  $a$  начиная с позиции  $j$  включительно.
- $f(i, x, y) = lcnt(i - 1, x) \cdot rcnt(i, y)$

Для каждого запроса необходимо по всем  $j$  от 2 до  $n$  найти максимум  $f(j, x_i, y_i)$ . Так как ученики пропустили все занятия, они не могут решить задачу ГРИБ'а. Помогите ученикам школы Центра Помощи Мигрантам решить эту задачу и избежать отчисления из школы.

### Формат входных данных

В первой строке даны два целых числа  $n$  и  $m$  ( $2 \leq n \leq 100\,000, 1 \leq m \leq 100\,000$ ) — количество чисел в массиве и число запросов.

Во второй строке даны  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^9$ ) — числа в массиве.

В следующих  $m$  строках описаны запросы. В каждой из них даны два целых числа  $x_i$  и  $y_i$  ( $1 \leq x_i, y_i \leq 10^9$ ) — значения из  $i$ -го запроса. Гарантируется, что числа  $x_i$  и  $y_i$  присутствуют в массиве.

### Формат выходных данных

В  $m$  строках выведите ответы на запросы, по одному в строке.

### Примеры

стандартный ввод	стандартный вывод
5 3 1 2 3 2 1 1 2 2 2 1 2	2 1 2
5 4 1 1 1 2 2 1 1 1 2 2 2 2 1	2 6 1 0

### Замечание

Рассмотрим первый пример.  
Первый запрос — 1 2:

- $f(2, 1, 2) = 2$
- $f(3, 1, 2) = 1$
- $f(4, 1, 2) = 1$
- $f(5, 1, 2) = 0$

Таким образом, ответ на первый запрос равен 2.

Второй запрос – 2 2:

- $f(2, 2, 2) = 0$
- $f(3, 2, 2) = 1$
- $f(4, 2, 2) = 1$
- $f(5, 2, 2) = 0$

Таким образом, ответ на второй запрос равен 1.

Третий запрос совпадает с первым, и ответ на него равен 2.

## Система оценки

Тесты к этой задаче состоят из 5 групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов некоторых из предыдущих групп. Обратите внимание, прохождение тестов из условия не требуется для некоторых групп. **Offline-проверка** означает, что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.

Группа	Баллы	Доп. ограничения			Необх. группы	Комментарий
		$n$	$m$	$a_i$		
0	0	–	–	–	–	Тесты из условия.
1	14	$n \leq 100$	$m \leq 100$	–	0	
2	19	$n \leq 5000$	$m \leq 5000$	–	0, 1	
3	22	–	–	$a_i \leq 1000$	–	
4	12	–	–	–	–	$x_i = y_i$ во всех запросах
5	33	–	–	–	0 – 4	<b>Offline-проверка</b>

## Задача F. Австралийская ПСП

Имя входного файла:	стандартный ввод или <code>input.txt</code>
Имя выходного файла:	стандартный вывод или <code>output.txt</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Как известно, в Австралии смотрят на вещи под самыми разными углами, поэтому и правильные скобочные последовательности они задают нестандартным способом:

- Пустая скобочная последовательность считается правильной.
- Если  $S$  считается правильной, то  $)S($ ,  $(S)$ ,  $[S]$ ,  $]S[$ ,  $\{S\}$ ,  $\}S\{$ ,  $\langle S \rangle$  и  $\rangle S \langle$  тоже считаются правильными.
- Если  $S$  и  $T$  считаются правильными, то  $S + T$  тоже считается правильной (здесь  $+$  означает конкатенацию строк).

Мальчик Вася решил посетить Австралию. Но вот беда, для этого надо пройти австралийский тест на интеллект! В самом сложном задании теста даётся строка  $s$ , состоящая из скобок и к ней даются  $m$  заданий двух видов:

1. Заменить скобку на позиции  $a_i$ .
2. Сказать, считается ли подстрока  $s$  на позициях с  $l_i$  по позицию  $r_i$  включительно правильной скобочной последовательностью в Австралии.

Вася очень просит вас помочь ему пройти тест.

### Формат входных данных

Первая строка содержит единственное целое число  $n$  ( $1 \leq n \leq 200\,000$ ) — длину скобочной последовательности.

Во второй строке содержится строка  $s$  длины  $n$ , состоящая из скобок `()[]{}<>` — исходная строка, данная Васе.

В третьей строке содержится целое число  $m$  ( $1 \leq m \leq 200\,000$ ) — количество заданий теста.

В следующих  $m$  строках заданы запросы. В  $i$ -й из следующих строк в начале содержится целое число  $t_i$  ( $1 \leq t_i \leq 2$ ).

- Если  $t_i = 1$ , то далее строка содержит целое число  $a_i$  и символ  $c_i$  ( $1 \leq a_i \leq n$ ). В этом случае требуется в строке  $s$  на позиции  $a_i$  заменить скобку на  $c_i$ . Гарантируется, что  $c_i$  является одной из скобок `()[]{}<>`.
- Если  $t_i = 2$ , то далее строка содержит два целых числа  $l_i$  и  $r_i$  ( $1 \leq l_i \leq r_i \leq n$ ). В этом случае требуется узнать, считается ли подстрока  $s$  на позициях с  $l_i$  по позицию  $r_i$  правильной скобочной последовательностью в Австралии.

### Формат выходных данных

Для каждого запроса второго типа выведите «Yes» (без кавычек), если скобочная последовательность считается правильной и «No» (без кавычек) иначе.

## Примеры

стандартный ввод	стандартный вывод
6 )() (()	Yes Yes
7 2 1 6 1 4 )	Yes No
2 2 5 1 3 [ 1 4 ] 2 1 6 2 4 5	
10 >() ( [ < } { 6 2 1 10 1 3 ( 2 1 10 2 2 5 1 2 ) 2 1 10	Yes No No Yes

## Замечание

В первом примере:

1. В первом задании просят сказать, считается ли подстрока «)() (())» правильной. Ответ «Yes», так как «)()» и «()» считаются правильными, а эта подстрока представляется, как сумма таких строк.
2. Во втором задании просят заменить скобку на 4-й позиции на ). После этого строка будет равна «)() (())».
3. В третьем задании просят сказать, считается ли подстрока «() (())» правильной. Ответ «Yes» аналогично первому заданию.
4. В четвёртом задании просят заменить скобку на 3-й позиции на [. После этого строка будет равна «) ( [ ( ) ».
5. В пятом задании просят заменить скобку на 4-й позиции на ]. После этого строка будет равна «) ( [ ( ) ».
6. В шестом задании просят сказать, считается ли подстрока «) ( [ ( ) » правильной. Ответ «Yes» аналогично предыдущим заданиям, так как « [ ] » тоже считается правильной.
7. В седьмом задании просят сказать, считается ли подстрока «] ( ) » правильной. Но нетрудно убедиться, что правильной она не считается, поэтому ответ «No».

Во втором примере:

1. В первом задании просят сказать, считается ли подстрока «>() ( [ < } {» правильной. Ответ «Yes», так как «()» и «>() (» считаются правильными, поэтому их сумма «>() ( [ < } {» считается правильной, поэтому строка «>() ( [ < } {» считается правильной. Также «} {» считается правильной, поэтому исходная подстрока считается правильной.
2. Во втором задании просят заменить скобку на 3-й позиции на (. После этого строка будет равна «>() ( [ < } {».

3. В третьем задании просят сказать, считается ли подстрока «>() [] [<]{» правильной. Ответ «No», так как иначе строка «() (» считалась бы правильной, но нетрудно убедиться, что это не так.
4. В четвёртом задании просят сказать, считается ли подстрока «() (» правильной. Нетрудно убедиться, что правильной она не считается, поэтому ответ «No»
5. В пятом задании просят заменить скобку на 2-й позиции на ). После этого строка будет равна «>) () [] [<]{».
6. В шестом задании просят сказать, считается ли подстрока «>) () [] [<]{» правильной. Ответ «Yes», так как «) () (» правильная, поэтому и «>) () (<» правильная, поэтому и исходная подстрока правильная.

## Система оценки

Тесты к этой задаче состоят из 6 групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов некоторых из предыдущих групп. Обратите внимание, прохождение тестов из условия не требуется для некоторых групп. **Offline-проверка** означает, что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.

Группа	Баллы	Доп. ограничения			Необх. группы	Комментарий
		$n$	$m$	$t_i$		
0	0	–	–	–	–	Тесты из условия.
1	16	$n \leq 100$	$m \leq 100$	–	0	
2	15	$n \leq 10\,000$	$m \leq 10\,000$	–	0, 1	
3	12	$n \leq 10\,000$	–	$t_i = 2$	–	
4	13	–	–	–	–	В любой момент строка состоит только из скобок ( и )
5	20	–	–	$t_i = 2$	3	
6	24	–	–	–	0 – 5	<b>Offline-проверка.</b>

## Задача G. Очередная скобочная последовательность

Имя входного файла:	стандартный ввод или <code>input.txt</code>
Имя выходного файла:	стандартный вывод или <code>output.txt</code>
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Вам дан массив  $a$ , состоящий из  $n$  целых чисел  $a_1, a_2, \dots, a_n$ . Назовём *стоимостью* правильной скобочной последовательности  $s$  длины  $n$  сумму чисел в массиве  $a$ , которые стоят на позициях, где в  $s$  стоят открывающие скобки «(».

Ваша задача состоит в том, чтобы найти правильную скобочную последовательность длины  $n$ , стоимость которой максимальна.

В этой задаче правильная скобочная последовательность — это последовательность, которую можно построить по следующим правилам:

- Пустая последовательность является правильной скобочной последовательностью;
- Если  $A$  — правильная скобочная последовательность, то последовательность  $(A)$  является правильной скобочной последовательностью;
- Если  $A$  и  $B$  — правильные скобочные последовательности, то последовательность  $AB$  (соединение этих последовательностей) является правильной скобочной последовательностью.

Например, последовательности  $(())()$ ,  $()$  и  $((()))$  являются правильными, а  $()($ ,  $((()$  и  $((()))($  не являются.

### Формат входных данных

Каждый тест состоит из нескольких наборов входных данных. Первая строка содержит единственное целое число  $t$  ( $1 \leq t \leq 10^5$ ) — количество наборов входных данных. Далее следует описание наборов входных данных.

Первая строка каждого набора входных данных содержит единственное четное целое число  $n$  ( $2 \leq n \leq 2 \cdot 10^5$ ) — длину массива  $a$ .

Вторая строка каждого набора входных данных содержит  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^9$ ) — элементы массива  $a$ .

Гарантируется, что сумма  $n$  по всем наборам входных данных не превосходит 200 000.

### Формат выходных данных

Для каждого набора входных данных выведите единственную строку  $s$  — правильную скобочную последовательность длины  $n$  с максимальной возможной стоимостью. Если правильных ответов несколько, выведите любой из них.

### Пример

стандартный ввод	стандартный вывод
5	()
2	((()))
1 10	()()
4	((()))()((()))
4 7 1 1	((()))()
4	
1 1 5 3	
10	
1 2 1 1 1 2 2 2 1 1	
6	
3 3 1 3 4 4	

## Замечание

В первом наборе входных данных существует единственная правильная скобочная последовательность длины 2 — «()». Её стоимость равна 1, потому что  $a_1 = 1$ .

Во втором наборе входных данных есть две правильные скобочные последовательности длины 4 — «(())» и «()()». У первой стоимость равна  $a_1 + a_2 = 4 + 7 = 11$ , у второй стоимость равна  $a_1 + a_3 = 4 + 1 = 5$ , поэтому ответ равен «(())».

В четвёртом наборе входных данных ответом является «(())()()()», стоимость которой равна  $a_1 + a_2 + a_5 + a_6 + a_7 = 1 + 2 + 1 + 2 + 2 = 8$ .

## Система оценки

Тесты к этой задаче состоят из 4 групп, не считая тесты из условия. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов необходимых групп. Обратите внимание, прохождение тестов из условия не требуется для некоторых групп.

Обозначим за  $N$  сумму  $n$  по всем тестовым набором внутри одного теста.

Группа	Баллы	Доп. ограничения		Необх. группы	Комментарий
		$N$	$a_i$		
0	0	–	–	–	Тесты из условия.
1	23	$N \leq 16$	–	0	
2	17	$N \leq 1000$	–	0, 1	
3	24	–	$a_i \leq 2$	–	
4	36	–	–	0, 1, 2, 3	

## Задача Н. Выращивание кроликов

Имя входного файла:	стандартный ввод или <code>input.txt</code>
Имя выходного файла:	стандартный вывод или <code>output.txt</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Разочаровавшись в олимпиадной информатике после школьного этапа, маленький мальчик Захар решил заняться разведением кроликов. Для этого он купил себе  $n$  кроликов и пронумеровал их по неубыванию весов. Изначально  $i$ -й кролик весит  $w_i$  килограмм.

Захар заметил, что если кроликов выстроить в шеренгу в порядке неубывания весов, то каждый день у каждого кролика, такого что его вес меньше веса следующего, вес увеличивается на 1 килограмм. Более формально, каждый день вес  $i$ -го кролика увеличивается на 1 кг, если он не последний в шеренге, и  $w_i \neq w_{i+1}$ . Все такие изменения происходят одновременно для всех кроликов. Такое происходит до тех пор, пока веса всех кроликов не становятся равными весу последнего кролика в шеренге.

Захар очень любознательный мальчик, а поэтому ему хочется ответить на  $m$  запросов: через сколько дней вес кроликов больше не будет меняться, если в качестве шеренги взять кроликов с номерами от  $l_i$  до  $r_i$ . Так как Захар разочарован в олимпиадной информатике, помочь ему предстоит вам.

### Формат входных данных

В первой строке задано единственное целое число  $n$  ( $1 \leq n \leq 200\,000$ ) — количество кроликов у Захара.

В следующей строке заданы  $n$  целых чисел  $w_1, w_2, \dots, w_n$  ( $1 \leq w_i \leq 10^9$ ) — веса кроликов. Гарантируется, что эти числа отсортированы по неубыванию, то есть  $w_i \leq w_{i+1}$ , для всех  $i \leq n - 1$ .

В следующей строке задано единственное целое число  $m$  ( $1 \leq m \leq 200\,000$ ) — количество запросов.

В следующих  $m$  строках задано по два целых числа  $l_i$  и  $r_i$  ( $1 \leq l_i \leq r_i \leq n$ ) — границы шеренги из  $i$ -го запроса.

### Формат выходных данных

На каждый запрос в отдельной строке выведите количество дней, после которых веса кроликов не будет меняться. Можно показать, что такой день обязательно наступит.

### Пример

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

### Замечание

В первом запросе после первого дня веса были равны 1, 3, 3, 7, после второго [2, 3, 4, 7], третьего [3, 4, 5, 7], [4, 5, 6, 7], [5, 6, 7, 7], [6, 7, 7, 7] и после седьмого [7, 7, 7, 7]

### Система оценки

Тесты к этой задаче состоят из 6 групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов некоторых из предыдущих групп. Обратите внимание, прохождение тестов из условия не требуется для некоторых групп. **Offline-проверка** означает, что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.



Длинный тур отборочного этапа Открытой олимпиады школьников 2022–2023 учебного года  
Россия. 21 ноября 2022 – 15 января 2023

Группа	Баллы	Доп. ограничения			Необх. группы	Комментарий
		$n$	$m$	$w$		
0	0	–	–	–	–	Тесты из условия.
1	14	$n \leq 100$	$m \leq 100$	$w \leq 100$	0	
2	17	$n \leq 500$	$m \leq 500$	$w \leq 500$	0, 1	
3	23	$n \leq 10\,000$	$m \leq 10\,000$	–	0, 1, 2	
4	12	$n \leq 100\,000$	$m \leq 100\,000$	$w \leq 2$	–	
5	13	$n \leq 100\,000$	$m \leq 100\,000$	$w \leq 1000$	0, 1, 2, 4	
6	21	–	–	–	0 – 5	<b>Offline-проверка</b>

## Задача I. Гладкие числа

Имя входного файла: стандартный ввод или `input.txt`  
Имя выходного файла: стандартный вывод или `output.txt`  
Ограничение по времени: 4 секунды  
Ограничение по памяти: 1024 мегабайта

Число называется  $b$ -гладким, если все его простые делители не превышают  $b$ . Число  $x$  называется простым делителем числа  $y$ , если  $y$  делится нацело на  $x$ ,  $x > 1$  и единственные два делителя числа  $x$  это 1 и  $x$ .

Даны  $n$  и  $b$ . Найдите количество  $b$ -гладких чисел от 1 до  $n$ .

### Формат входных данных

В единственной строке вводятся два целых числа  $n$  и  $b$  ( $4 \leq n \leq 10^{18}$ ,  $2 \leq b \leq 500$ ).

### Формат выходных данных

В единственной строке выведите количество  $b$ -гладких чисел от 1 до  $n$ .

### Примеры

стандартный ввод	стандартный вывод
7 2	3
12 3	8
10000 50	2463

### Замечание

Во втором примере 3-гладкими числами от 1 до 12 являются: 1, 2, 3, 4, 6, 8, 9, 12

### Система оценки

Тесты к этой задаче состоят из 18 групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов некоторых из предыдущих групп. Обратите внимание, прохождение тестов из условия не требуется для некоторых групп. **Offline-проверка** означает, что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.

Группа	Баллы	Доп. ограничения		Необх. группы	Комментарий
		$n$	$b$		
0	0	–	–	–	Тесты из условия.
1	7	$n \leq 10\,000$	$b \leq 100$	0	
2	8	$n \leq 10^7$	$b \leq 100$	0, 1	
3	8	$n \leq 10^{11}$	$b \leq 100$	0 – 2	
4	4	$n \leq 10^{11}$	$b \leq 300$	0 – 3	
5	6	$n \leq 10^{11}$	–	0 – 4	
6	12	$n \leq 10^{16}$	$b \leq 100$	0 – 3	
7	7	$n \leq 10^{12}$	–	0 – 5	
8	5	$n \leq 10^{13}$	–	0 – 5, 7	
9	5	$n \leq 10^{14}$	–	0 – 5, 7, 8	
10	9	–	$b \leq 100$	0 – 3, 6	
11	4	–	$b \leq 150$	0 – 3, 6, 10	
12	3	–	$b \leq 200$	0 – 3, 6, 10, 11	
13	3	–	$b \leq 250$	0 – 3, 6, 10 – 12	
14	3	–	$b \leq 300$	0 – 4, 6, 10 – 13	
15	4	$n \leq 10^{15}$	–	0 – 5, 7 – 9	
16	4	$n \leq 10^{16}$	–	0 – 9, 15, 16	
17	4	$n \leq 10^{17}$	–	0 – 9, 15 – 17	<b>Offline-проверка</b>
18	4	–	–	0 – 17	<b>Offline-проверка</b>