

Задача А. Найти максимум

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

Реализуйте структуру данных для эффективного вычисления максимума среди нескольких подряд идущих элементов массива.

Формат входных данных

В первой строке вводится одно натуральное число N ($1 \leq N \leq 100\,000$) — количество чисел в массиве.

Во второй строке вводятся N чисел от 1 до 100 000 — элементы массива.

В третьей строке вводится одно натуральное число K ($1 \leq K \leq 30\,000$) — количество запросов на вычисление максимума.

В следующих K строках вводится по два числа — номера левого и правого элементов отрезка массива (считается, что элементы массива нумеруются с единицы).

Формат выходных данных

Для каждого запроса выведите максимум среди всех чисел соответствующего участка массива. Числа выводите в одну строку через пробел.

Примеры

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

Задача В. Фонари

Имя входного файла:	<code>streetlamps.in</code>
Имя выходного файла:	<code>streetlamps.out</code>
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

В детском лагере «Берендеевы поляны» вряд ли найдётся кто-нибудь, кто не побоялся бы сходить ночью на хутор. В связи с этим администрация лагеря установила вдоль дороги N фонарей. Лампочки в фонарях могут перегорать, и, хотя в «Берендеевых полянах» есть электрик, он может заболеть и не успеть заменить перегоревшую лампочку на новую до приезда очередной комиссии.

Каждая посещающая лагерь комиссия считает своим долгом проинспектировать состояние фонарей, освещающих дорогу на хутор, но дорога эта очень длинная, поэтому каждый раз членов комиссии подвозят на машине к какому-нибудь фонарю, они выходят, идут пешком до другого фонаря, осматривая по пути все фонари, и там садятся в машину и уезжают.

Если они замечают хотя бы один перегоревший фонарь, в «Берендеевых полянах» устраивается штрафное посвящение для преподавателей.

По информации о происходящих событиях для каждого приезда комиссии выведите результат осмотра фонарей.

Формат входных данных

В первой строке входного файла записаны два числа N и k ($1 \leq N, k \leq 100\,000$) — количество фонарей и количество событий. Далее следуют k строк, описывающих события. События описываются следующим образом:

- перегорание фонаря задаётся двумя числами, первое из которых равно -1 , а второе задаёт номер перегоревшего фонаря (от 1 до N);
- замена лампочки электриком задаётся двумя числами, первое из которых равно 1, а второе задаёт номер заменённой лампочки (от 1 до N);
- приезд комиссии задается тремя числами, первое из которых равно 0, а два других (a и b) задают отрезок дороги, по которому проходит комиссия. А именно, члены комиссии видят все фонари с номерами от a до b включительно и только их. Гарантируется, что $1 \leq a \leq b \leq N$.

Изначально все фонари исправны.

Формат выходных данных

Для каждого приезда комиссии выведите в выходной файл одну строку. А именно, если члены комиссии не заметят ни одного перегоревшего фонаря, то выведите «PASSED», иначе — «PENALTY».

Пример

streetlamps.in	streetlamps.out
5 6	PASSED
-1 2	PENALTY
0 1 1	PASSED
0 2 5	
-1 4	
1 2	
0 2 3	

Задача С. Парковка

Имя входного файла: `parking.in`
Имя выходного файла: `parking.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

На кольцевой парковке есть n мест пронумерованных от 1 до n . Есть два вида событий прибытие машину на парковку и отъезд машины с парковки. Если машина приезжает на парковку, а её место занято, то она едет далее по кругу и встаёт на первое свободное место.

Формат входных данных

В первой строке входного файла находится два числа n и m — размер парковки и количество запросов ($1 \leq n, m \leq 100000$). В следующих m строках находятся события. Каждая из этих строк имеет следующий вид:

- `enter x` — приехала машина, которая хочет встать на место x . Для каждой такой команды выведите какое место займёт эта машина.
- `exit x` — уехала машина занимавшая место x . Гарантируется, что на этом месте была машина.

Формат выходных данных

Выведите последовательно результаты выполнения всех операций `enter`.

Примеры

<code>parking.in</code>	<code>parking.out</code>
<code>3 5</code>	<code>1</code>
<code>enter 1</code>	<code>2</code>
<code>enter 1</code>	<code>3</code>
<code>exit 1</code>	<code>1</code>
<code>enter 2</code>	
<code>enter 2</code>	

Задача D. Битовые OR и AND

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

Есть массив из n элементов, изначально заполненный нулями. Вам нужно написать структуру данных, которая обрабатывает два вида запросов:

- применить ко всем элементам от l до $r - 1$ операцию $a_i = a_i | v$ (битовый OR),
- найти битовый AND элементов на отрезке от l до $r - 1$.

Формат входных данных

Первая строка содержит два числа n и m ($1 \leq n, m \leq 100000$) — размер массива и число операций. Далее следует описание операций. Описание каждой операции имеет следующий вид:

- $1 \ l \ r \ v$ — применить ко всем элементам от l до $r - 1$ операцию $a_i = a_i | v$ ($0 \leq l < r \leq n$, $0 \leq v < 2^{30}$).
- $2 \ l \ r$ — найти битовый AND элементов на отрезке от l до $r - 1$ ($0 \leq l < r \leq n$).

Формат выходных данных

Для каждой операции второго типа выведите соответствующее значение.

Примеры

стандартный ввод	стандартный вывод
5 6 1 0 3 3 2 1 2 1 1 4 4 2 1 3 2 1 4 2 3 5	3 7 4 0
2 3 1 0 1 3 1 1 2 9 2 0 2	1

Задача E. RMQ Наоборот

Имя входного файла: `rmq.in`
 Имя выходного файла: `rmq.out`
 Ограничение по времени: 2 секунды
 Ограничение по памяти: 256 мегабайт

Рассмотрим массив $a[1..n]$. Пусть $Q(i, j)$ — ответ на запрос о нахождении минимума среди чисел $a[i], \dots, a[j]$. Вам даны несколько запросов и ответы на них. Восстановите исходный массив.

Формат входных данных

Первая строка входного файла содержит число n — размер массива, и m — число запросов ($1 \leq n, m \leq 100\,000$). Следующие m строк содержат по три целых числа i, j и q , означающих, что $Q(i, j) = q$ ($1 \leq i \leq j \leq n, -2^{31} \leq q \leq 2^{31} - 1$).

Формат выходных данных

Если искомого массива не существует, выведите строку `inconsistent`.

В противном случае в первую строку выходного файла выведите `consistent`. Во вторую строку выходного файла выведите элементы массива. Элементами массива должны быть целые числа в интервале от -2^{31} до $2^{31} - 1$ включительно. Если решений несколько, выведите любое.

Примеры

<code>rmq.in</code>	<code>rmq.out</code>
3 2 1 2 1 2 3 2	<code>consistent</code> 1 2 2
3 3 1 2 1 1 1 2 2 3 2	<code>inconsistent</code>

Задача F. Присваивание, прибавление и сумма

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	1024 мегабайта

Есть массив из n элементов, изначально заполненный нулями. Вам нужно написать структуру данных, которая обрабатывает три вида запросов:

- присвоить всем элементам на отрезке от l до $r - 1$ значение v ,
- прибавить ко всем элементам на отрезке от l до $r - 1$ число v ,
- узнать сумму на отрезке от l до $r - 1$.

Формат входных данных

Первая строка содержит два числа n и m ($1 \leq n, m \leq 100000$) — размер массива и число операций. Далее следует описание операций. Описание каждой операции имеет следующий вид:

- $1 \ l \ r \ v$ — присвоить всем элементам на отрезке от l до $r - 1$ значение v ($0 \leq l < r \leq n$, $0 \leq v \leq 10^5$).
- $2 \ l \ r \ v$ — прибавить ко всем элементам на отрезке от l до $r - 1$ число v ($0 \leq l < r \leq n$, $0 \leq v \leq 10^5$).
- $3 \ l \ r$ — узнать сумму на отрезке от l до $r - 1$ ($0 \leq l < r \leq n$).

Формат выходных данных

Для каждой операции третьего типа выведите соответствующее значение.

Примеры

стандартный ввод	стандартный вывод
5 7	8
1 0 3 3	10
2 2 4 2	4
3 1 3	
2 1 5 1	
1 0 2 2	
3 0 3	
3 3 5	