

## DIVMAC: Dividing Machine

*Легенда, описанная далее, переформулирована и упрощена переводчиком, чтобы читатель мог лучше понять условие задачи. Оригинальную легенду вы можете прочитать на странице задачи в контексте.*

### Условие:

Шеф придумал уникальную машину, которая умеет выполнять следующие операции с массивом целых неотрицательных чисел  $A$ :

- Операция 0:

**Update (L, R):**

```
for i = L to R:  
    A[i] = A[i] / LeastPrimeDivisor(A[i])
```

- Операция 1:

**Get (L, R):**

```
result = 1  
for i = L to R:  
    result = max(result, LeastPrimeDivisor(A[i]))  
return result
```

Здесь  $\text{LeastPrimeDivisor}(x)$  обозначает наименьший простой делитель числа  $x$ . Если у числа нету простых делителей, значение функции равно единице.

Шеф предоставляет вам массив из  $N$  элементов и  $M$  запросов на выполнение описанных выше операций. Вам нужно выполнить все эти операции и сказать Шефу ответ на каждую операцию типа 1.

### Формат ввода:

Первая строка содержит одно целое число  $T$  — количество тестов.

Первая строка каждого теста содержит два разделенных пробелом целых числа  $N$  и  $M$  — размер массива  $A$  и количество запросов.

Вторая строка содержит  $N$  разделенных пробелами целых чисел  $A_1, A_2, \dots, A_N$ .

Каждая из следующих  $M$  строк содержит три разделенных пробелами целых числа  $type, L$  и  $R$  — тип операции (0 — Update, 1 — Get) и ее параметры соответственно.

### Формат вывода:

Для каждого теста для каждого запроса типа 1 в отдельной строке выведите ответ на этот запрос.

# CODECHEF

---

## Ограничения:

- $1 \leq T \leq 100$
- $1 \leq A_i \leq 10^6$
- $1 \leq L \leq R \leq N$
- $0 \leq \text{type} \leq 1$
- Сумма всех  $M$  в одном тестовом файле не превышает  $10^6$
- **Подзадача 1**(10 баллов):  $1 \leq N, M \leq 10^3$
- **Подзадача 2**(25 баллов):  $1 \leq N, M \leq 10^5$ , все  $A_i$  — простые числа
- **Подзадача 3**(65 баллов):  $1 \leq N, M \leq 10^5$

## Примеры тестов:

### Входные данные:

```
1
6 7
2 5 8 10 3 44
1 2 6
0 2 3
1 2 6
0 4 6
1 1 6
0 1 6
1 4 6
```

### Выходные данные:

```
5 3 5 11
```

## Пояснение:

Состояния массива  $A$  после выполнения каждой **Update** операции:

- $A = [2\ 1\ 4\ 10\ 3\ 44]$
- $A = [2\ 1\ 4\ 5\ 1\ 22]$
- $A = [1\ 1\ 2\ 1\ 1\ 11]$