

Counting Primes

Tortoise and Achilles are playing the Counting the Primes game. Achilles will give Tortoise some numbers, and some intervals, and then Tortoise needs to count the primes on those intervals. It is an easy game, but Tortoise is doing the counting slowly. Achilles is pissed off, so he has given you the task as you are a good programmer. For a twist, he has changed the game a little bit, that is he will give some intervals for counting the prime as well as he will give some intervals to change the numbers in that interval.

You are given an array of n elements. After that you will be given M commands. They are:

- $0\ x\ y\ v$ - you have to change all numbers in the range of x to y (inclusive) to v , where x and y are two indexes of the array.
- $1\ x\ y$ - output a line containing a single integer which is the number of primes between x and y (inclusive).

The array is indexed from 1 to n .

Input

Input starts with an integer T (≤ 10), denoting the number of test cases.

Each case contains two integers n ($1 \leq n \leq 10^4$) and q ($1 \leq q \leq 2 \cdot 10^4$). Then next line, you will be given N integers. After that each of the next q lines will contain a task in one of the following form:

- $0\ x\ y\ v$ ($1 \leq x \leq y \leq n$, $2 \leq v \leq 10^6$)
- $1\ x\ y$ ($1 \leq x \leq y \leq n$)

And the numbers will be in range of $[2, 10^6]$.

Output

For each case, print the case number first. Then for each query ' $1\ x\ y$ ', print the number of primes between x and y [inclusively].

Example

Input:

```
1
5 3
78 2 13 12 3
1 1 2
0 4 4 5
1 1 5
```

Output:

```
Case 1:
1
4
```

Note:

- Use Faster IO like scanf, printf
- A prime number is a natural number greater than 1 that has no positive divisors other than 1 and itself. The first prime numbers are 2, 3, 5, 7, 11 ...