NOI '07 P4 - Necklace Factory

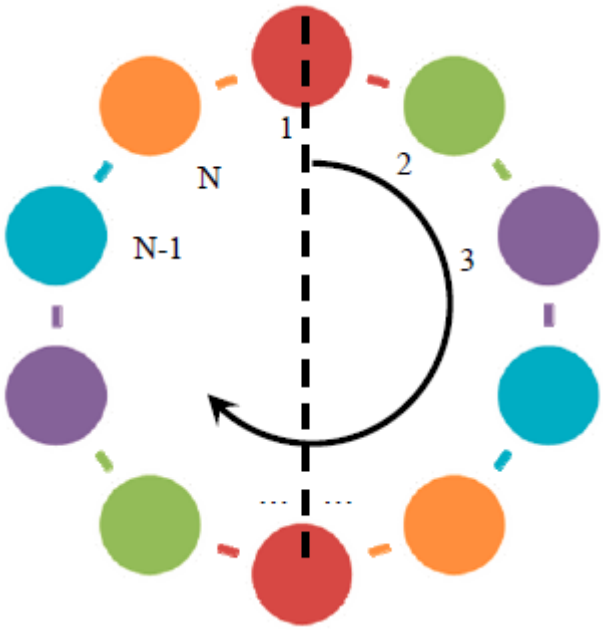
Time limit: 1.0s **Memory limit:** 256M

National Olympiad in Informatics, China, 2007

Company T specializes in selling necklaces with colored beads. The produced necklaces have fashionable designs, varied styles, and affordable prices, making them widely popular amongst young people. Recently, company T decided to launch a necklace "buffet" production system so that customers can decide for themselves which necklace design is most beautiful.

The necklace buffet production system includes a hardware system and a software system. The software system interacts with the user to control the hardware system, while the hardware system receives the commands from the software system and produces the specified necklace. The hardware system has already been completed, however the software system has yet to be developed. Company T's people have found you, competing at the National Olympiad in Informatics. Can you help company T write a software simulation system?

A necklace contains N beads, and each bead is a color from colors $1, 2, \ldots, c$. The necklace is secured onto a flat board. Some place on the board is labeled as position 1, and going clockwise from there, other places on the board are labeled positions $2, 3, \ldots, N$.



The program you design must support the following commands:

Command	Parameter Constraints	Details
`R k`	$0 < k < N$	Meaning *Rotate* k. The necklace on the board is rotated clockwise by k spots. The bead at position 1 moves to position $k+1$, the bead at position 2 moves to position $k+2$, and so on.

Command	Parameter Constraints	Details
`F`		Meaning *Flip*. The necklace is flipped along the given axis of symmetry. The bead at position 1 does not move, the bead at position 2 swaps places with the bead at position N, the bead at position 3 swaps places with the bead at position $N-1$, and so forth.
`S i j`	$1 \leq i, j \leq N$	Meaning *Swap* i, j. The bead at position i swaps places with the bead at position j.
`P i j x`	$1 \leq i, j \leq N$; $x \leq c$	Meaning *Paint* i, j. The segment of beads starting at position i, inclusive, and ending at position j, inclusive, is painted color x.
`C`		Meaning *Count*. Query how many "sections" form the current necklace. We consider a consecutive segment with all beads the same color to be a "section".
`CS i j`	$1 \leq i, j \leq N$	Meaning *CountSegment* i, j. Query how many sections form the segment of the current necklace starting at position i, inclusive, and ending at position j, inclusive.

Input Specification

The first line of input contains two integers N and c, respectively representing the number of beads in the necklace and the number of colors.
The second line contains N integers $x_1, x_2, \ldots, x_N$, representing the color of the beads from locations 1 through N, where $1 \leq x_i \leq c$.
The third line of input contains an integer Q, the number of queries to follow.
The following Q lines each contain a single command in one of the formats described above.

Output Specification

For each `C` and `CS` command, output on a separate line an integer corresponding to the appropriate answer.

Sample Input

```
5 3
1 2 3 2 1
4
C
R 2
P 5 5 2
CS 4 1
```

Sample Output

```
4
1
```

Constraints

For 60% of the test cases: $N \leq 1\,000$, $Q \leq 1\,000$.
For 100% of the test cases: $N \leq 500\,000$, $Q \leq 500\,000$, $c \leq 1\,000$.

Clarifications

Regarding Rotations and Flips

Note that the rotate command rotates the *beads*, not the numberings of the *positions*. Also, the flip command will always have position 1 as its axis of symmetry.

For example, when $N = 10$, the position numberings are depicted in *Fig. 1* below:

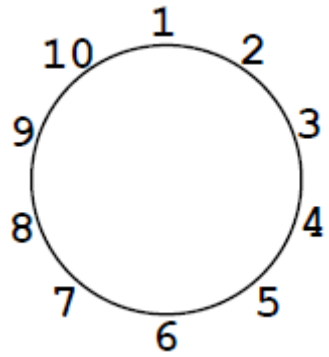


Fig. 1: The numbering of bead positions.

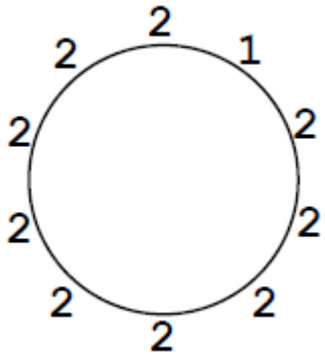


Fig. 2: The initial colors of the beads.

Assuming the necklace beads are initially colored like *Fig. 2*, we can say that only the bead at position 2 is colored 1. The beads at all other positions are colored 2.

After performing an `R 2` command, the colors of the beads will be as depicted in *Fig. 3* below:

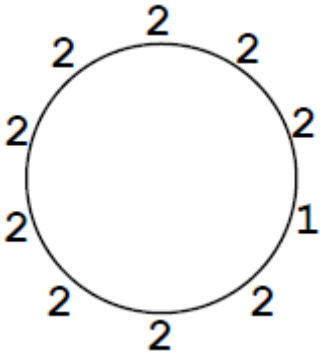


Fig. 3: Bead colors after carrying out `R 2`.

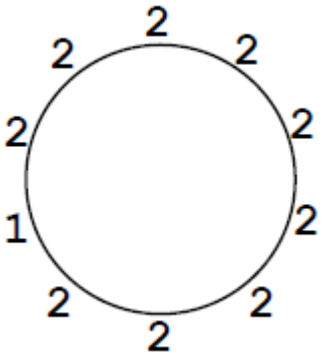


Fig. 4: Bead colors after carrying out `F`.

Note that currently, the numbering of necklace positions still remains the same as in *Fig. 1*, so the axis for flipping still remains unchanged. Thus, after making an `F` command, the bead colorings will be as depicted in *Fig. 4*.

Regarding the CountSegment Command

The `CS` command asks for how many "sections" are within a "segment". Especially take note of when a length N segment is queried. We still have to treat the queried region as a "segment".

Take the scenario in *Fig. 4* for example. Carrying out a `CS 1 10` command, the query asks for how many "sections" make up the segment starting at position 1 and ending at position 10. This should result in an output of 3. In comparison, just carrying out a `C` command would result in an output of 2.

Problem translated to English by **Alex**.