beCP
2023
Task 2.3: Password (password)

Author: Damien Galant
Preparation: Damien Galant
Time limit: 1 s Memory limit: 512 MB

Since you're training for the International Olympiad in Informatics, your computer scientist friend kindly agrees to give you access to his online collection of C++ algorithms. It is protected by a password. In order to challenge you, your friend suggests that you find the password yourself by solving one of his algorithmic problems. Are you up to his challenge?

His problem concerns the strings formed by the 26 letters of the alphabet, in lowercase (for example `becp`).

Two operations are considered:

1. Remove all occurrences of the letter l_1 in the string.
2. Replace all occurrences of the letter l_1 by the letter l_2 in the string.

For instance, if the operation 1 is applied to the string `aeeeeaecae`, with $l_1 =$ `a`, the resulting string is `eeeece`.

If the 2 operation is applied to the string `abbbabcaba` with $l_1 =$ `b` and $l_2 =$ `e`, the resulting string is `aeeeeaecaea`.

Your friend tells you how to find the password to his collection of algorithms:

- A string of S characters is given to you. This is the initial string.
- A sequence of M operations of type 1 and 2 is given.
- K times in a row, you must apply the M operations, in the order in which they are provided. If the operations are $O_1, \cdots, O_M$, then you successively apply

$$\underbrace{O_1, O_2, \cdots, O_M}_{\text{1st iteration}}, \underbrace{O_1, O_2, \cdots, O_M}_{\text{2th iteration}}, \cdots, \underbrace{O_1, O_2, \cdots, O_M}_{k\text{th iteration}}.$$

  From S, you apply O_1 and get a string S_1, then apply O_2 to S_1 and get a string S_2, and so on.

The string obtained after these $K \cdot M$ operations (i.e. $S_{K \cdot M}$) is the password you are looking for. Now, it is up to you!

Input

The first line contains the three integers N, M and K. The second line contains the initial string S, which only contains lowercase letters. The M following lines are of one of the two following types:

- 1 l_1, describing an operation of type 1;
- 2 l_1 l_2, describing an operation of type 2.

Output

The output consists in a single line containing a string made of lowercase letters: your friend's password. Is is guaranteed that the password to guess is nonempty.

General limits

- S is made of N lowercase letters among `a`–`z`;
- for each operation of the type 2, we have $l_1 \neq l_2$;
- $1 \leq N$;
- $1 \leq M$;
- $N \cdot M \leq 10^6$;
- $1 \leq K \leq 10^{18}$.

Additional constraints

Subtask	Points	Constraints
A	15	$K = M = 1$
B	20	There are only operations of type 1
C	20	$K \cdot N \cdot M \leq 10^6$
D	20	There are only operations of type 2
E	25	No additional constraint

Important remark: some numbers may exceed the capacity of a 32-bit integer, so use the type `long long`.

Example 1

sample1.in

```
10 1 1
aeeeaecaea
1 a
```

sample1.out

```
eeeece
```

This example may appear in subtasks A, B, C and E.

Example 2

sample2.in

```
10 1 1
abbbabcaba
2 b e
```

sample2.out

```
aeeeaecaea
```

This example may appear in subtasks A, C, D and E.

Example 3

sample3.in

```
10 3 2
uvbwewucvp
2 u v
1 v
2 w u
```

sample3.out

```
becp
```

This example may appear in subtasks C and E. In this example, we have to repeat twice a succession of three operations, of types 2, 1 and 2. The intial string is `uvbwewucvp`. By applying the operations, we obtain successively `vvbwewvcvp`, `bwewcp`, `bueucp`, `bvevcp`, `becp` and `becp`. The password is thus `becp`!