Palindromic Subsets

Consider a lowercase English alphabetic letter character denoted by c. A *shift* operation on some c turns it into the next letter in the alphabet. For example, and $shift(\mathtt{a}) = \mathtt{b}$, $shift(\mathtt{e}) = \mathtt{f}$, $shift(\mathtt{z}) = \mathtt{a}$.

Given a zero-indexed string, s, of n lowercase letters, perform q queries on s where each query takes one of the following two forms:

- `1 i j t`: All letters in the inclusive range from i to j are shifted t times.
- `2 i j`: Consider all indices in the inclusive range from i to j. Find the number of non-empty subsets of characters, $c_1, c_2, \ldots, c_k$ where $i \leq$ index of $c_1 <$ index of $c_2 < \ldots <$ index of $c_k \leq j)$, such that characters $c_1, c_2, c_3, \ldots, c_k$ *can be rearranged* to form a palindrome. Then print this number modulo $10^9 + 7$ on a new line. Two palindromic subsets are considered to be different if their component characters came from different indices in the original string.

Note Two palindromic subsets are considered to be different if their component characters came from different indices in the original string.

Input Format

The first line contains two space-separated integers describing the respective values of n and q.
The second line contains a string of n lowercase English alphabetic letters (i.e., `a` through `z`) denoting s.
Each of the q subsequent lines describes a query in one of the two formats defined above.

Constraints

- $1 \leq n \leq 10^5$
- $1 \leq q \leq 10^5$
- $0 \leq i \leq j < n$ for each query.
- $0 \leq t \leq 10^9$ for each query of type 1.

Subtasks

For 20% of the maximum score:

- $n \leq 500$
- $q \leq 500$

For another 30% of the maximum score:

- All queries will be of type 2.

Output Format

For each query of type 2 (i.e., `2 i j`), print the number of non-empty subsets of characters satisfying the conditions given above, modulo $10^9 + 7$, on a new line.

Sample Input 0

```
3 5
aba
2 0 2
2 0 0
2 1 2
1 0 1 1
2 0 2
```

Sample Output 0

```
5
1
2
3
```

Explanation 0

We perform the following $q = 5$ queries:

1. `2 0 2`: $s = \textbf{aba}$ and we want to find the palindromic subsets of substring $\textbf{aba}$. There are five such subsets that form palindromic strings (**a**, **b**, **a**, **aa**, and **aba**), so we print the result of $5 \bmod (10^9 + 7) = 5$ on a new line
2. `2 0 0`: $s = \textbf{aba}$ and we want to find the palindromic subsets of substring $\textbf{a}$. Because this substring only has one letter, we only have one subset forming a palindromic string (**a**). We then print the result of $1 \bmod (10^9 + 7) = 1$ on a new line.
3. `2 1 2`: $s = \textbf{aba}$ and we want to find the palindromic subsets of substring $\textbf{ba}$. There are two such subsets that form palindromic strings (**b** and **a**), so we print the result of $2 \bmod (10^9 + 7) = 2$ on a new line.
4. `1 0 1 1`: $s = \textbf{aba}$ and we need to perform $t = 1$ shift operations on each character from index $i = 0$ to index $j = 1$. After performing these shifts, $s = \textbf{bca}$.
5. `2 0 2`: $s = \textbf{bca}$ and we want to find the palindromic subsets of substring $\textbf{bca}$. There are three valid subsets that form palindromic strings (**b**, **c**, and **a**), so we print the result of $3 \bmod (10^9 + 7) = 3$ on a new line.