

# Repeating Subsequence Tests

어떤 문자열의 부분열(subsequence)은, 문자열에서 몇몇 문자를 지웠을 때 (혹은 하나도 지우지 않아도 되고, 모두 지워도 된다.) 얻어지는 문자열이다. 예를 들어, “ababA”의 서로 다른 부분열은 “”(빈 문자열), “a”, “b”, “A”, “ab”, “aa”, “aA”, “ba”, “bb”, “bA”, “aba”, “abb”, “abA”, “aab”, “aaA”, “bab”, “baA”, “bbA”, “abab”, “abaA”, “abbA”, “aabA”, “babA”, “ababA”로 총 24개다. 이 문제에서는 대소문자를 구별하는 것에 유의하라.

길이가  $N$ 인 문자열  $S$ 가 주어진다. 이 때, 이 문자열의 연속한 부분문자열  $Q$  개에 대해 각각의 서로 다른 부분열의 개수를 구하는 프로그램을 작성하라.

입력을 받는 시간을 줄이기 위해 다음과 같은 의사 난수 생성기(Pseudo-Random Generator)를 사용한다.  $a_0, b_0, p, q, r$  이 주어지면,  $i \geq 1$  일 때  $a_i, b_i, x_i, y_i$ 는 다음과 같이 정의된다. 먼저  $X = 1,000,000,007$ 라고 하자.

$$\begin{aligned} a_i &= (p \cdot a_{i-1} + q \cdot b_{i-1} + r) \bmod X \\ b_i &= (p \cdot b_{i-1} + q \cdot a_{i-1} + r) \bmod X \\ x_i &= \min(a_i \bmod N, b_i \bmod N) + 1 \\ y_i &= \max(a_i \bmod N, b_i \bmod N) + 1 \end{aligned}$$

$u \bmod v$ 는  $u$ 를  $v$ 로 나눈 나머지를 의미한다. 이 때,  $i$ 번째 부분 문자열은  $S$ 의  $x_i$ 번째 문자에서  $y_i$ 번째 문자 까지를 포함하는 문자열이다. 의사 난수 생성기가 난수를 생성하는 방식에는 의미가 없으며, 임의의 입력이 주어져도 적절한 시간 내에 정답을 출력해내는 알고리즘이 존재한다.

## 입력

첫 번째 줄에 문자열  $S$ 가 주어진다. 이 문자열의 길이  $N$ 은 1이상  $10^6$ 이하이며,  $S$ 는 알파벳 대문자, 소문자만으로 구성되어 있다.

두 번째 줄에는 6개의 정수  $Q, a_0, b_0, p, q, r$ 이 공백 하나로 구분되어 주어진다.  $1 \leq Q \leq 10^6$ 을 만족하며,  $0 \leq a_0, b_0, p, q, r < X = 1,000,000,007$ 을 만족한다.

## 출력

출력이 오래 걸릴 수 있기 때문에,  $i$ 번째 부분문자열의 서로 다른 부분열의 개수를  $z_i$ 라고 할 때,  $z_i \bmod X (= 1,000,000,007)$ 를 모두 bitwise-XOR하여 첫 번째 줄에 출력하도록 한다.

## 채점 기준

$N \leq 10^5, Q \leq 10^5$  인 범위의 입력만 해결하면 2점을 획득할 수 있다.

모든 입력을 해결하면 7점을 획득할 수 있다.

입력 예제	출력 예제
ababA 2 4 6 8 10 12	16

$x_1 = 1, y_1 = 5$ 로 계산되어, 문자열 “ababA”의 서로 다른 부분열의 개수를 계산해야 하며, 위에서 구했듯  $z_1 = 24$ 이다.

$x_1 = 3, y_1 = 5$ 로 계산되어, 문자열 “abA”의 서로 다른 문자열의 개수를 계산해야 하며, 문자들이 서로 다르기 때문에 중복이 없어  $z_2 = 8$ 이다.

두 수를 1,000,000,007로 나눈 나머지를 bitwise-XOR하면 16이 되므로, 16을 출력하면 된다.