Let's define a function, f, on a string, p, of length l as follows:

$$f(p) = (p_1 \cdot a^{l-1} + p_2 \cdot a^{l-2} + \cdots + p_l \cdot a^0) \bmod m$$

where p_i denotes the ASCII value of the i^{th} character in string p, $a = 100001$, and $m = 10^9 + 7$.

Nikita has a string, s, consisting of n lowercase letters that she wants to perform q queries on. Each query consists of an integer, k, and you have to find the value of $f(w_k)$ where w_k is the k^{th} alphabetically smallest palindromic substring of s. If w_k doesn't exist, print -1 instead.

Input Format

The first line contains 2 space-separated integers describing the respective values of n (the length of string s) and q (the number of queries).
The second line contains a single string denoting s.
Each of the q subsequent lines contains a single integer denoting the value of k for a query.

Constraints

- $1 \leq n, q \leq 10^5$
- $1 \leq k \leq \frac{n \cdot (n+1)}{2}$
- It is guaranteed that string s consists of lowercase English alphabetic letters only (i.e., `a` to `z`).
- $a = 10^5 + 1$
- $m = 10^9 + 7$.

Scoring

- $1 \leq n, q \leq 10^3$ for 25% of the test cases.
- $1 \leq n, q \leq 10^5$ for 100% of the test cases.

Output Format

For each query, print the value of function $f(w_k)$ where w_k is the k^{th} *alphabetically smallest palindromic substring* of s; if w_k doesn't exist, print -1 instead.

Sample Input

```
5 7
abcba
1
2
3
4
6
7
8
```

Sample Output

```
97
97
696207567
98
29493435
99
-1
```

Explanation

There are 7 palindromic substrings of `"abcba"`. Let's list them in lexicographical order and find value of w_k:

1. $w_1 =$ `"a"`, $f(w_1) = 97$

2. $w_2 =$ `"a"`, $f(w_2) = 97$
3. $w_3 =$ `"abcba"`, $f(w_3) = 696207567$
4. $w_4 =$ `"b"`, $f(w_4) = 98$
5. $w_5 =$ `"b"`, $f(w_5) = 98$
6. $w_6 =$ `"bcb"`, $f(w_6) = 29493435$
7. $w_7 =$ `"c"`, $f(w_7) = 99$
8. $w_8 =$ doesn't exist, so we print -1 for $k = 8$.