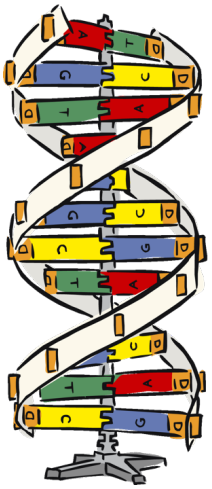
DNA



A DNA sequence consists of a combination of four different letters: A, T, C, and G. We are interested in the number of occurrences of some parts of the DNA in a given DNA sequence. That is, given a sequence of a DNA of length **N**, you need to check how many times a given subsequence of a DNA of length **K** appears in the DNA sequence.

Input

The first line of the input contains **N** (1 <= N <= 500) and **K** (1 <= K <= 6), where N is the length of the DNA sequence and K is the length of the subsequence that we are interested in. The second line contains the DNA sequence of length N. The next line contains an integer **Q** (1 <= Q <= 1,000), denoting the number of subsequences that we are interested in. The next Q lines contain the subsequences of length K.

Output

For each subsequence of the DNA, print the number of occurrences of that subsequence in the DNA sequence given to you. Print them in separate lines and the last line of the output should contain a newline character.

Sample Input

```
6 2
ACGTAC
6
AC
CG
AT
GT
TA
CA
```

Sample Output

```
2
1
0
1
1
0
```

Explanation

ACGTAC: There are **2 AC** in our DNA sequence.
ACGTAC: There is **1 CG** in our DNA sequence.
ACGTAC: There is **no AT** in our DNA sequence.
ACGTAC There is **1 GT** in our DNA sequence.
ACGTAC There is **1 TA** in our DNA sequence.
ACGTAC There is **no CA** in our DNA sequence.

Skeleton

You are given the skeleton file **`DNA.java`**

Note

1. Your solution must be able to answer each query in **O(K)** time to pass all the given test cases, OK?