DNA Searching

Problem Description

We are interested in the number of occurrences of some parts of DNA in a given DNA sequence. Given a sequence of DNA, find the number of occurrences of those substrings in that DNA.

Input

The first line of the input contains N ($1 <= N <= 500$) and K ($1 <= K <= 6$), where N is the length of that DNA and K is the length of the substring 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 substrings that we are interested in. The next Q lines contain the substrings of length K.

Output

There are Q lines in the output. Each line contains the number of occurrences of the substrings.

Sample Input

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

Sample Output

Sample Output	Explanation	
2	**AC**GT**AC**	There are 2 **AC** in our DNA sequence
1	A**CG**TAC	There is 1 **CG** in our DNA sequence
0	ACGTAC	There is 0 **AT** in our DNA sequence
1	AC**GT**AC	There is 1 **GT** in our DNA sequence
1	ACG**TA**C	There is 1 **TA** in our DNA sequence
0	ACGTAC	There is 0 **CA** in our DNA sequence

Marking

1. You will only gain a maximum of 70% for this section if you solution is worse than O(N.K + Q.K).
2. You can gain a maximum of 100% for this section if O(N.K + Q.K) or better solution is implemented correctly. You must answer each query in O(K) time.

Hint

- To get 100% solution, think 1 data structure that you have learnt that can search/query a particular key in O(1).

Note

The main Java class must be called **DNA**, and be in the source file **DNA.java**.