Problem D. Demolition Time

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

Darkwing Duck is in trouble again! As soon as he returned to his town, it was announced that Negaduck has been attacking the place. As a proper villain, he is going to rob some set of **consecutively** placed buildings and escape with the loot.

The panorama of the town can be represented as a string s where each character represents the shape of the corresponding building. Different buildings may have the same shape, and hence are represented in the string by equal characters. Negaduck has picked some string p as his robbery plan. Fortunately, this string p has become known to the police. Now they want to find all the buildings which are in danger, but there is one complication on the way.

By the order of the new Mayor, some of the buildings in the town should be demolished. They are demolished one after another **in the order they appear** in the string s: from left to right. This demolition plan is well known in the city.

The robbery can take place anytime: before all planned demolitions, after them, or between the demolition of any two buildings which follow one another in the demolition plan. However, no demolitions can happen during the robbery.

Negaduck robs buildings strictly according to his robbery plan from left to right. He does not skip any buildings except for demolished ones. He can execute only the whole plan. Nevertheless, there can exist a lot of subsets of buildings which are in danger of being robbed.

Formally, consider string t we get from s by removing the characters corresponding to already demolished buildings. Note that string t is initially equal to s but changes after each demolition. Negaduck can rob a subset of buildings if there exists a moment of time when this subset forms a substring of t, and this substring is equal to string p.

Your task is to find the exact amount of subsets which are in danger of being robbed.

Input

The first line of input contains a non-empty string s consisting of lowercase and uppercase English letters, digits, characters '`,`', '`!`', '`_`', '`.`' and '`-`' representing the buildings in the town from left to right. The length of this string doesn't exceed 1 000 000.

The second line of input contains a string p consisting of lowercase and uppercase English letters, digits, characters '`,`', '`!`', '`_`', '`.`' and '`-`' representing the Negaduck's plan from left to right. The length of this string doesn't exceed 1 000 000.

Note that uppercase and lowercase English letters are considered different.

The third line of input contains one integer m ($0 \le m \le |s|$) — the length of the Mayor's demolition plan.

The next line contains an **increasing** sequence of integers $x_1, x_2, \ldots, x_m$ ($1 \le x_1 < \ldots < x_m \le |s|$) — the indices of buildings in order they should be demolished.

Output

Output one integer: the number of different subsets of buildings that can be robbed according to Negaduck's plan.

Example

standard input	standard output
aabbcc abc 3 2 4 5	2

Note

The answer for the given sample is 2 because the following possible sequences of buildings can be robbed: 1 3 5 (after two building demolitions) and 1 3 6 (after all three building demolitions).