

CFG

Input file: **standard input**
Output file: **standard output**
Time limit: 0.5 seconds
Memory limit: 128 megabytes

Consider a function F , which takes in a string and outputs a string, having the following properties:

- For every string x , $|x| < |F(x)|$.
- For every strings x and y , $F(xy) = F(x)F(y)$, where xy is x concatenated with y .

We will also construct a set of strings, L , in the following manner:

- B is in L .
- If x is in L then $F(x)$ is also in L .
- If x and y are both in L then xy is also in L .

You are given a string B and a list, A , of N strings. Knowing that B is in L , determine for each string from A if it belongs to L or not.

You will have to write a function with the following header:

```
string check(string (*F)(string), string B, vector<string>A)
```

The function should return a string, where the i – *th* character is '1' if the i – *th* string from A belongs to L and '0' otherwise.

Note that you only have to implement the check function, but you can also implement additional helping functions and / or structures or classes.

Please refer to the sample implementation (sample.cpp).

Input

The check function gets the following three arguments (in this particular order):

- The F function.
- The string B .
- The array of strings A .

All the strings will only contain lowercase English alphabet letters.

For tests worth 10 points: the length of the longest string from $A \leq 10$.

For tests worth 30 more points: the length of the longest string from $A \leq 1000$.

For tests worth 10 more points: B only contains 'a's and F only returns strings containing 'a's.

For tests worth 50 more points: $|B| \leq 10^5$, the length of the longest string from $A \leq 10^5$.

For every testcase, A contains at most 20 strings.

Output

The check function returns a string of length equal to the length of A , where the i – *th* character is '1' or '0' depending on whether or not the string $A[i]$ is in L .

Note

Given its special type, this problem doesn't have a formal sample test case with a sample output. Here we present and explain the sample test.

Let's consider $B = ab$ and $F(B) = abbac$. Then:

- ab is in L .
- $abab$ is in L .
- $abbac$ is in L .
- $ababbac$ is in L .
- $abbc$ is not in L .
- abb is not in L .
- cd is not in L .
- ...