

Problem A. Ciphertext

Input file: *standard input*
Output file: *standard output*
Time limit: 1 second
Memory limit: 256 mebibytes

Stierlitz wants to deliver a very important message s to the headquarters. He uses a prefix code defined in the National Standard. Unfortunately, the enemy knows the code, and the communication channel is tapped. In order to avoid suspicion, Stierlitz wants to split the ciphertext into several parts in such a way that none of the parts is a valid encoding of some string using the code. In order to maximize security, Stierlitz wants to maximize the number of parts that he will split the ciphertext into. Find this maximum number of parts, or determine that it is impossible to split the ciphertext satisfying the requirements.

Input

The first line of input contains one integer k ($1 \leq k \leq 52$): the size of the alphabet. The possible characters are enumerated from 1 to 52 in A-Za-z order, and the text of the message will only contain characters with numbers from 1 to k .

The second line of input contains a non-empty string s of size up to 10^6 which consists of characters with indices from 1 to k (see enumeration rules described above).

The next k lines contain binary codes of alphabet characters according to the enumeration order. Each character is encoded by a non-empty sequence of 0s and 1s of length at most k . It is guaranteed that no character has a code that is equal to the prefix of the code for a different character.

Output

Print a single integer: the maximum number of parts, or -1 in case it is impossible to split the ciphertext into parts satisfying the requirements.

Examples

| standard input | standard output |
|-----------------------------------------|-----------------|
| 3 CACB 011 1 001 | 2 |
| 3 ACBABCAACABCAACC 0 10 110 | -1 |

Note

In the first example, the encoded text looks like 0010110011. The only way to split it into parts which are not valid encodings of any strings is 0 010110011. Therefore, the answer is 2.

In the second example, it is possible to prove by induction that any string ending with 0 and not having three 1s in a row is a valid encoding of some string. Every suffix of the encoded text matches this criterion, therefore the answer is -1 .