Morphisms

The objects to be studied in this assignment are called morphisms. Suppose Σ is a finite set of characters, which we will assume is a subset of $\{a, b, c, d, e, f\}$. Then Σ^* denotes the set of all sequences whose elements are these characters, that is, it is just the set of all strings, also called words, containing only the given set of characters.

A morphism f is a function $f : \Sigma \to \Sigma^*$ that assigns to each character $x \in \Sigma$ some word $f(x) \in \Sigma^*$. The morphism f can be extended to all words in Σ^* by defining $f(\lambda) = \lambda$ and $f(w \cdot x) = f(w) \cdot f(x)$, for all words $w \in \Sigma^*$ and characters $x \in \Sigma$. Here λ is the empty word and $\cdot$ denotes concatenation.

Suppose $f(a)$ also has a as the first character. Then consider the sequence of words $a, f(a), \ldots, f^n(a), \ldots$, where $f^0(a) = a$ and $f^n(a) = f(f^{n-1}(a))$. It can be seen that $f^{n-1}(a)$ is a prefix of $f^n(a)$ for all $n \geq 1$, and as $n \to \infty$ this sequence approaches a fixed, possibly infinite, word called its limit.

You have to write a program to find some properties of this sequence.

1. Given a number n, find the length of $f^n(a)$.

2. Given a number i, find the character at index i in the limit, or equivalently, the character at index i of $f^n(a)$ for any n for which length of $f^n(a)$ is greater than i. The 0th character will always be a. It can be assumed that i is less than the length of the limit.

3. Given a word w, determine if it is substring of $f^n(a)$ for some n, and if so find the smallest such n. If not a substring for any n, output -1. If the word is a substring, find the smallest index in the limit at which it occurs as a substring.

4. Given a word w, determine if it is a subsequence of $f^n(a)$ for some n, and if so, find the smallest possible n. Again, output -1 if it is not a subsequence. If it is a subsequence, also output the length of the shortest prefix of the limit, such that w is a subsequence of the prefix.

Input/Output The first line of input will specify the number of characters k, where $1 \leq k \leq 6$. It can be assumed that the characters are the first k characters in the sequence a, b, c, d, e, f. The next k lines will contain a string each, the ith line giving the value of f for the ith character. It can be assumed that the length of each string is at most 20. The next line will specify the number of test cases. Each test case, given in a separate line, will consist of one of the 4 operations defined above. The type of an operation is defined by one of the numbers $0, 1, 2, 3$, which will be given first on the line. If the operation is 0 or 1, it will be followed by a single non-negative integer n. If the operation is 0, the output should be the length of $f^n(a)$ and if the operation is 1, the output is the character at index n in the infinite sequence. It can be assumed the length of $f^n(a)$ will be at most 10^{18} for operation 0, and n will be at most 10^{18} for operation 1. If the operation is 2 or 3, this will be followed by a string of length at most 1000. If

the operation is 2, the output should be the smallest n such that the given string is a substring of $f^n(a)$, followed by the smallest index at which it occurs, and -1 if it not a substring. For operation 3, the output should be the smallest n such that the given string is a subsequence of $f^n(a)$, and the length of the shortest prefix of $f^n(a)$ of which it is a subsequence, and -1 if there is no such n. The output of each test case must be printed on a separate line, in the same order as the input.

Note: Two of the most well-studied morphisms are the Fibonacci morphism defined by $f(a) = ab$ and $f(b) = a$, and the Thue-Morse morphism $f(a) = ab$ and $f(b) = ba$. The infinite words generated by them are called the Fibonacci and Thue-Morse words. You will get half credit if you solve the problems for these two special cases. If you choose to do only this, you can first check if the input morphism is one of these, and if so solve the problem, else do nothing. All 4 parts will be checked independently and carry equal marks.

Homework: Find all strings w such that $wi \cdot w$ is a substring of the Fibonacci word, and also strings w such that $w \cdot w \cdot w$ is a substring. Does there exist a string w such that $w \cdot w \cdot w \cdot w$ is a substring of the Fibonnaci word? Prove that the Fibonacci word contains excatly $n+1$ distinct substrings of length n. Prove that the Thue-Morse word is overlap-free, that is, it does not contain two overlapping occurrences of the same string. This implies it is cube-free. Can you find the number of distinct substrings of length n of the Thue-Morse word. Construct a morphism with 3 letters such that the generated word is square-free.

Submission: Submit a single file named RollNo_2.cpp .