Can Cat match the pattern?

Last week, thanks to you, Cat was able to attend her Bioinformatics midterm. This week, she is working on her term project for the Bioinformatics class. She needs a pattern-matching program to find a specific subsequence of a given sequence so that she can identify the structures she is studying. To help her, you offer to write a pattern-matcher. Given a sequence of symbols and a pattern to search for in the sequence your program should match all overlapping and non-overlapping occurrences of the pattern, and report the starting indices of the occurrences.

There is one twist: two different symbols can be valid in each other's places. For example, given the mapping T <-> U and the pattern AGTU, we can conclude that AGTU, AGUU, AGTT, and AGUT are all valid patterns.

To be more specific, you are given a string *sequence* of length N, a *mapping* between two symbols, and a *pattern* of length M. You should scan the *sequence* and find the occurrences of the given *pattern*. You are expected to print the starting indices of each occurrence of the pattern, in increasing order.

You are expected to implement a **Finite State Automaton Matcher**. The construction of FSA can be implemented without any optimization and can even run in $O(M^4)$ time, but, as expected, the FSA matching algorithm should run in linear time (i.e. O(N)) by making exactly one comparison for each character in the text. ***Even if you pass the test cases and time limits, solutions that do not obey these specifications will get a 0 grade.***

For this THE, as we expect you to implement the algorithmic details, you are not allowed to use any advanced algorithm libraries. For this reason, **your code is not allowed to include any additional headers/ libraries** other than the ones that we already provide you.