We say that a number N is *weakly divisible* by an integer k if N is divisible by k, or removal of one of the digits of N gives an integer that it divisible by k. As an example, consider 653. Although it is not divisible by 7, removal of 5 gives 63 which is divisible by 7. Thus, 653 is weakly divisible by 7. Another larger example is 7418426079388661994435796800837062546488295193992 68 - a 50 digit number that is NOT weakly divisible by 7. This means none of the 51 numbers (the number itself and the 50 numbers you get by deleting one of the 50 digits are not divisible by 7).

The goal of the project is to determine, for two given integers N and k, the number of k digit integers that are weakly divisible by k. For example, when $N = 7$ and $k = 2$, the two digit numbers that are weakly divisible by 7 are 10, 14, 17, 20, 21, 27, 28, 30, 35, 37, 40, 42, 47, 49, 50, 56, 57, 60, 63, 67, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 84, 87, 90, 91, 97, 98 so the output is 37.

Algorithm to solve this problem: Given a positive integer N, we know how to build a DFA that accepts the decimal representation of all the integer multiples of N. Recall that this DFA M has n states - one state for each remainder $0, 1, \cdots, N-1$. We used such a DFA in Problem 2 of Project 1. It is not hard to modify the idea behind this DFA construction to build an NFA with $2N$ states that accepts the decimal representation of integers that are weakly divisible by N. The idea is as follows: Make two copies of the DFA M, and add transitions on all inputs $0, 1, \cdots, 9$ from state i of copy 1 to state i of the second copy, for all i. Call this NFA M_1. The figure below shows the NFA that accepts all the (positive) integers that are weakly divisible by 3. (This NFA accepts integers that begin with 0 such as 039, but in your construction, the NFA should reject such strings.) The next step is to convert M_1 to a DFA M_2 using subset construction. The final step is to count the number of strings of length k accepted by M_2