1   Congruence Mod 7 in Base 10

Recall that in order to find a 'trick' for divisibility by 7, we take powers of 7 until we see a pattern:

All $\equiv$ are mod 7

$10^0 \equiv 1$

$10^1 \equiv 10 \equiv 3$

$10^2 \equiv 10 \times 10 \equiv 3 \times 3 \equiv 9 \equiv 2$

$10^3 \equiv 10^2 \times 10 \equiv 2 \times 3 \equiv 6$

$10^4 \equiv 10^3 \times 10 \equiv 6 \times 3 \equiv 18 \equiv 4$

$10^5 \equiv 10^4 \times 10 \equiv 4 \times 3 \equiv 12 \equiv 5$

$10^6 \equiv 10^5 \times 10 \equiv 5 \times 3 \equiv 15 \equiv 1$

Ah, so the pattern is:

$(1, 3, 2, 6, 4, 5, 1, 3, 2, 6, 4, 5, \cdots)$

Hence, the way to tell what the remainder is when dividing:

$a_n a_{n-1} a_{n-2} \cdots a_0$ is divided by 7 is to compute, mod 7,

$1 \times a_0 + 3 \times a_1 + 2 \times a_2 + 6 \times a_3 + 4 \times a_4 + 5 \times a_5 +$

$1 \times a_6 + 3 \times a_7 + 2 \times a_8 + 6 \times a_9 + 4 \times a_{10} + 5 \times a_{11} + \cdots +$

Hence,

- The number of states in the DFA classifier for mod 7 is $6 \times 7 = 42$.

- The weights you would use for the DFA are $(1, 3, 2, 6, 4, 5)$.

2 Congruence Mod 4 in Base 10

Recall that in order to find a 'trick' for divisibility by 4, we take powers of 4 until we see a pattern:

All $\equiv$ are mod 4.

$10^0 \equiv 1$

$10^1 \equiv 10 \equiv 2$

$10^2 \equiv 10 \times 10 \equiv 2 \times 2 \equiv 4 \equiv 0$

$10^3 \equiv 10^2 \times 10 \equiv 0 \times 2 \equiv 0$

$10^4 \equiv 10^3 \times 10 \equiv 0 \times 2 \equiv 0$

$10^5 \equiv 10^4 \times 10 \equiv 0 \times 2 \equiv 0$

$10^6 \equiv 10^5 \times 10 \equiv 0 \times 2 \equiv 0$

Ah, so the pattern is:

$(1, 2, 0, 0, 0, 0, 0, \cdots)$

Hence, the way to tell what the remainder is when dividing:

$a_n a_{n-1} a_{n-2} \cdots a_0$ is divided by 4 is to compute, mod 4,

$1 \times a_0 + 2 \times a_1$

We first look at a naive but correct way to make a DFA for this.

- We need to keep track of the weighted sum mod 4. There are 4 possibilities.

- We need to keep track of whether this is the kind of state where when we see a digit we use weight 1 OR weight 2 OR weight 0. There are 3 possibilities.

- Hence we can build a DFA with $3 \times 4 = 12$ states.

I know what you are saying: This is B*LLSH*T MAN! In class we did it with just 9 states.

Given a base b and a congruence M finding the pattern for $b^n \pmod{M}$ is easy. Finding the weights is easy. Finding the number of states for the NAIVE DFA is easy. Finding the DFA with the minimal number of states seems, if not hard, harder than I want for this project. However, it will be extra credit as you will see.

3 What to do in General

What if you wanted to do divisibility by M in base b.

All $\equiv$ are mod M.

1. Compute b^0 (mod M), b^1 (mod M), b^2 (mod M),, b^3 (mod M), $\cdots$ until you find a pattern.

2. For the number of states you need to look carefully at if the first few digits matter (as they did for base 10, divisible by 4). With this in mind you can find the number of states

3. You have the weights.

4 Your Project

Write a program that will do the following:

Input: b and M. Both are natural numbers ≥ 2.

Output:

- The weights.

- The number of states in the naive DFA classifier for mod M in base b.

Name your program (it should be a single file): lastnamefirstinitial

Your program should take two arguments from the command line, b and M and output the *unique* weights (no repetition), in the correct order, as a JSON compatible list (i.e. a list with items separated by comma and enclosed with square brackets) as the first line of output and the number of states in the naive DFA classifier as the second line of output. For example, in base 10, for divisibility by 7:

```
$ ./gasarchb 10 7
[1, 3, 2, 6, 4, 5]
42
```

If you're writing your code in a scripting language (e.g. python2, python3, perl, ruby, bash), put a shebang (`https://en.wikipedia.org/wiki/Shebang_(Unix)`) as the *first line* of the file:

```
#!/usr/bin/env python3
```

If you want to use a compiled language (Why do you want to use a compiled language!?), or a different scripting language than what's above, talk to Josh Brulé (the TA) first.

Regardless of the language you're writing in, put a comment near the top of your code (but *not* as the first line; the first line should be the shebang) that has your full name and UID.

When you're confident your program is correct, submit your program to the submit server (`https://submit.cs.umd.edu/`)

5   Extra Credit

Given b and M what is the size of the MINIMAL DFA-classifier? One COULD obtain the DFA and then minimize it. This would not be satisfying.

For Extra Credit (goes towards a letter I might write for you, not towards the grade) think brilliant thoughts on this problem and tell me about them and see if you can code it up.