Modular arithmetic

(exercises)

Application of discrete models

1. A Hungarian bank account number contains either 16 or 24 digits, split into 8-digit groups (xxxxxxxx-xxxxxxxx or xxxxxxxx-xxxxxxxx-xxxxxxxx). Each correct account number must satisfy the following rule: if the digits are multiplied by $1, 3, 7, 9, 1, 3, 7, 9, \ldots$ respectively, then the sum of these products must be a multiple of 10. This is true both for the first 8 digits and for the remaining 8 or 16 digits.
   a) Write a function that decides whether the given bank account number is correct (according to this rule).
   b) Is 10402196-50526765-81771006 correct? If not, which part has an error?
   c) Can we tell which digit is wrong?
   d) Write a function that receives a (potentially incorrect) bank account number and the index of the single wrong digit, and it returns the correct number.

2. Write a function that solves the linear congruence $ax \equiv b \pmod{m}$ for given $a, b \in \mathbb{Z}$ and $m \in \mathbb{N}^+$. The return value is the list of all solutions between 0 and $m-1$. Solve it in two ways:
   a) using brute force search;
   b) using the learnt method. (Don't use Sage's equation solving functions, but its other functions can be used.)
   c) Solve the congruence $30517578125x \equiv 33729082272 \pmod{999999999999}$. Which method works?

3. Write a function that calculates the multiplicative inverse of $a \in \mathbb{Z}_m$ for a given $m \in \mathbb{N}^+$. Don't use Sage's equation solving functions and modular arithmetic, and don't use brute force search. The input and ouput numbers are integers (not residue classes). If the inverse doesn't exist, return `None`. Calculate the inverse of 123456789 modulo 79590490282399.

4. Implement the Chinese remainder theorem for two congruences, i.e. write a function that finds the common solutions of the congruences $x \equiv a \pmod{m}$ and $x \equiv b \pmod{n}$. Expect the same four parameters as `crt()`, but unlike `crt()`, return the solution as a residue class, using the proper modulus. If there is no solution, return `None`. (Don't use `crt()` or Sage's other equation solving functions.)