Elementary Number Theory

Definition: Given $a, b \in \mathbb{Z}$ not simultaneously zero their *greatest common divisor*, denoted $\gcd(a, b)$ or (a, b) is the largest integer dividing both a and b. If $\gcd(a, b) = 1$ then a and b are said to be *relatively prime* or *coprime*.

Proposition: *Given a and b as before, there exist $x, y \in \mathbb{Z}$ such that*

$$ax + by = \gcd(a, b).$$

Sketch of the proof: Use Euclidean algorithm. i.e., note that $a = bc + r \Rightarrow \gcd(a, b) = \gcd(b, r)$ and iterate this fact. □

IMPORTANT REMARK: The computation of $\gcd(a, b)$ and the computation of x and y require a quantity of operations comparable to the number of digits. This is very quick for a computer with the numbers employed in actual cryptography (hundreds of digits).

Sage commands: `gcd(a,b)` or `gcd([list])`. The extended version `xgcd(a,b)` gives the gcd and x and y in the proposition.

```
sage: gcd(18,42)
6
sage: gcd([18,42,14])
2
sage: xgcd(18,42)
(6, -2, 1)
```

Definition: We say that *a is congruent to b modulo $m \in \mathbb{Z}^+$*, denoted $a \equiv b \pmod{m}$ or $a \equiv b\ (m)$, if $m \mid a - b$.

Recall: the symbol $\mid$ means "divides". Its negation is $\nmid$.

For a fixed m the congruence defines an equivalence relation. The classes are denoted by $\mathbb{Z}/m\mathbb{Z}$. This set inherits the $+$ and $\times$ operations from $\mathbb{Z}$. Note that $\overline{a} \in \mathbb{Z}/m\mathbb{Z}$ represent the set of all number differing from a in a multiple of m. In some cases we replace the heavy notation $\overline{a}$ by a.

Proposition: *An element* $\overline{a} \in \mathbb{Z}/m\mathbb{Z}$ *has a multiplicative inverse if and only if* $\gcd(a, m) = 1$.

Sketch of the proof: Note that $1 = ax + my$ means $\overline{a} \cdot \overline{x} = \overline{1}$ in $\mathbb{Z}/m\mathbb{Z}$. □

Sage commands: `n.inverse_mod(m)`. In some sense `Mod(a,n)` means $\overline{a}$.

```
sage: 10.inverse_mod(13)
4
sage: Mod(4*10,13)
1
sage: Mod(4,13)*Mod(10,13)
1
```

Definition: The elements of $\mathbb{Z}/m\mathbb{Z}$ having multiplicative inverse are called *units*. The Euler ϕ-function is the function that assigns to each m the number of units in $\mathbb{Z}/m\mathbb{Z}$.

Fact: It is not difficult to prove that if $m = p_1^{\alpha_1} p_2^{\alpha_2} \cdots p_k^{\alpha_k}$ is the prime factorization of m then

$$\phi(m) = \prod_{i=1}^{k} p_i^{\alpha_i - 1}(p_i - 1) = m \prod_{p|m} \left(1 - \frac{1}{p}\right).$$

Sage commands: `euler_phi(m)`. This function involves the factorization of m hence it is in general very hard for the computer when m has hundreds of digits.

```
sage: euler_phi(39)
24
sage: time euler_phi(10^120+1)
CPU times: user 1494.33 s, sys: 1.20 s, total: 1495.53 s
Wall time: 1504.81 s
```

Proposition (Chinese Remainder Theorem): *Given* m_1 *and* m_2 *relatively prime, there exists* x *such that*

$$\begin{cases} x \equiv a_1 \pmod{m_1} \\ x \equiv a_2 \pmod{m_2} \end{cases}$$

Indeed x *is unique if we impose* $0 \leq x < m_1 m_2$.

Sketch of the proof: Consider $x = a_1 M_2 m_2 + a_2 M_1 m_1$ where M_1 is the inverse of m_1 modulo m_2 and M_2 is the inverse of m_2 modulo m_1. □

Sage commands: `crt(a_1,a_2,m_1,m_2)`.

```
sage: crt(4,6,7,11)
39
sage: Mod(39,7)
4
sage: Mod(39,11)
6
```

A little of Algebra (see actual definitions in your favorite book):
An abelian group is a set G with an operation that behaves as the addition in $\mathbb{Z}$, i.e, satisfies commutativity, associativity, existence of identity element and existence of inverse element.

The units of $\mathbb{Z}/m\mathbb{Z}$ form a group with respect to multiplication, the group of units. The entire set $\mathbb{Z}/m\mathbb{Z}$ is an abelian group when endowed with addition.

One of the most basic theorems in group theory is *Lagrange Theorem*, saying that if $H \subset G$ are finite groups (with the same operation) then $\#H$ divides $\#G$.

In a finite abelian group the powers of an element (repeated operation of an element with itself) form a group. Its cardinality is called the *order* of the element. In other words the order is the minimal $k \in \mathbb{Z}^+$ such that it takes k self-operations over a to reach the identity element.

Proposition (Euler-Fermat congruence): *Let a and m be relatively prime. then*

$$a^{\phi(m)} \equiv 1 \pmod{m}.$$

Proof: According to Lagrange Theorem applied to the units of $\mathbb{Z}/m\mathbb{Z}$ we have $a^k \equiv 1 \pmod{m}$ for some k dividing the cardinality of the group which is $\phi(m)$. □

For p prime the units of $\mathbb{Z}/p\mathbb{Z}$ are all the classes except $\overline{0}$ then the previous result reads

$$a^{p-1} \equiv 1 \pmod{p} \qquad \text{for every } p \nmid a.$$

This is called *Fermat's little theorem.*

Definition: It can be proved (it is elementary but not simple) that there are some elements $\overline{a} \in \mathbb{Z}/p\mathbb{Z}$ such that $\{\overline{a}^1, \overline{a}^2, \ldots, \overline{a}^{p-1}\} = \mathbb{Z}/p\mathbb{Z} - \{\overline{0}\}$. An element $\overline{a}$ or a with this property is called a *primitive root* modulo p.

Sage commands: The order of a unit in $\overline{a} \in \mathbb{Z}/m\mathbb{Z}$ can be computed with `Mod(a,m).multiplicative_order()`. The command `primitive_root(p)` generates a primitive root modulo p.

```
sage: Mod(2,5).multiplicative_order()
4
sage: Mod(2^4,5)
1sage: print [Mod(2^i,5) for i in range(12)]
[1, 2, 4, 3, 1, 2, 4, 3, 1, 2, 4, 3]
sage: primitive_root(5)
2
sage: primitive_root(103)
5
```

A computational short cut to calculate power modulo m is

1. Use Euler-Fermat congruence if possible.
2. Employ the repeated squaring method.

The latter method consists of writting the exponent in base-2 system to reduce all the calculations to squaring over and over.

For instance to compute $x \equiv 2011^{196} \pmod{127}$ note firstly $2010 \equiv 106 \equiv -21$, then $x \equiv 21^{196}$. Euler-Fermat congruence says $21^{126} \equiv 1$ then $x \equiv 21^{70}$. Writting $70 = 2^6 + 2^2 + 2$ we have $x \equiv 21^{2^6} \cdot 21^{2^2} \cdot 21^2$ that can be computed by iterated squaring of 21 (recall to reduce modulo 127 after each squaring).

Sage commands: Computing $a^n \pmod{m}$ is easy for the computers even when large numbers (of hundred of digits) are involved in the calculation. The most direct command is `power_mod(a,n,m)`.

```
sage: power_mod(5^40+2,7^30,10^20+1)
65622873844338379843
sage: # Also
sage: Mod(5^40+2,10^20+1)^(7^30)
65622873844338379843
sage: # Avoid this way of doing the calculation!!!
sage: Mod((5^40+2)^(7^30),10^20+1)
```