This is not mathematically rigorous or complete.

1 Useful Definitions

- For integers a and b, $a \geq b$, the greatest common divisor is written as $gcd(a, b)$.
- Two numbers a and b are **relatively prime** or **coprime** if $gcd(a, b) = 1$.
- If $a = qn + r$ where $0 \leq r < n$, then **Remainder** $r = a \pmod{n}$
- A **multiplicative inverse** for x is a y such that $x.y = 1$.
- A **modulo multiplicative inverse** for an integer a is denoted by a^{-1} such that $aa^{-1} \equiv 1 \pmod{n}$. It exists only if a and n are coprime. As an example, the multiplicative inverse for $8 \pmod{11}$ satisfies $1 \equiv 8(7)(\text{mod } 11)$. Euclidean algorithm can be used to compute the inverse.

2 Encryption/Decryption Example

- Pick two prime numbers $p = 11$ and $q = 17$. $n = p * q$ =187.
- Compute Euler's Totient functions: $\phi(p) = (p - 1)$ =10, $\phi(q) = (q - 1)$ =16, $\phi(n) = \phi(p) * \phi(q)$ =160, $\phi(187)$ =160.
- Pick a number $e$between 1 and $\phi(187)$ that is coprime with 160 i.e. e and $\phi(187)$ have no common factors. Let e =19.
- Compute the modulo multiplicative inverse d. It is $d = 59$. $1 \equiv 19(59)(\text{mod } 160)$
- (n, e) is one key, its counterpart is (n, d).
- Let $m = 42$ be the message to be encrypted. Encrypt the message: $c = m^e \pmod{n}$. This is the message to be tranmitted. $c = 42^{19} \bmod 187 = 104$.
- The recipient gets c and decrypts it using (n, d). $m = c^d \pmod{n}$. $m = 104^{59} \pmod{187} = 42$

- There is nothing special about e and d. What is encrypted with d can be decrypted by e. This works because of the property: $m = (m^e)^d (\text{mod } n)$ and $m = (m^d)^e (\text{mod } n)$

3 Digital Signature

- Given message m compute $hash(m)$.
- Encrypt the $hash(m)$ with d to create signature s, send (m, s) to recipient(s).
- A recipient can verify the signature by hashing the message m, decrypting the signature s with e and comparing the hashes.

4 Euclid's Algorithm for computing gcd

For integers a and b, $a \geq b$, $gcd(a, b) = gcd(b, a - bq)$. Using this repetitively yields the $gcd(a, b)$.

$a = q_0 b + r_0$
$b = q_1 r_0 + r_1$
$r_0 = q_2 r_1 + r_2$
$r_1 = q_3 r_2 + r_3$

The remainders r_k steadily decrease, eventually going to zero for r_N, then $gcd(a, b) = r_{N-1}$.

Example

Let $a = 1071$, $b = 462$

$1071 = q_0 462 + r_0$ ($q_0 = 2$, $r_0 = 147$)
$462 = q_1 147 + r_1$ ($q_1 = 3$, $r_1 = 21$)
$147 = q_2 21 + r_2$ ($q_2 = 7$, $r_3 = 0$)

$gcd(1071, 462) = 21$

The following python code shows a simple implementation.

```
cat > gcd.py<<EOF
#! /usr/bin/python
from __future__ import print_function
def gcd(a, b):
  if a < b:
    a, b = b, a
  r = a % b
  while r !=0:
    a, b = b, r
    r = a % b
  return b
print (gcd(1071,462))
EOF

$python gcd.py
21
```

5 Congruence Relationship

Two integers a and b are said to be **congruent modulo** n, written as:

$a \equiv b \pmod{n}$

Remainders of integer division of both a and b by n are the same. Alternately $(a - b)$ is an integer multiple of n.

This notation is equivalent to:

$a \pmod{n} = b \pmod{n}$

Example $a = 38$, $b = 14, n = 12$

$38 \equiv 14 \pmod{12}$

The remainder of 38/12 and 14/12 is 2. Alternatively (38 - 24) is divisible by 12.

The value $y \equiv a^x (\text{mod } n)$ can be efficiently computed even when a^x is large and y can be computed without dealing with numbers larger than n^2.

6 Fermat Little Theorem

Fermat's little theorem states that for every prime number p and every integer a:

$a^p \equiv a \ (\text{mod } p)$

If a is not divisible by p, it is equivalent to:

$a^p - 1 \equiv 1 \ (\text{mod } p)$

Lemma

$(x + y)^p (\text{mod } p) = x^p + y^p (\text{mod } p)$

A (mod p) operation on binomial expansion of left hand side leaves only 2 terms. The remainder of rest of the terms after dividing by p is zero. In congruent modulo notation:

$(x + y)^p (\text{mod } p) \equiv x^p + y^p (\text{mod } p)$.

Proof of Fermat's little Theorem

The proof is by induction. Assume $k^p \ (\text{mod } p) = k \ (\text{mod } p)$ is true.

Consider:

$(k + 1)^p \equiv (k + 1)^p (\text{mod } p)$

From Lemma:

$(k + 1)^p \equiv k^p + 1^p (\text{mod } p)$

Using induction with $1^p = 1$ being trivially true:

$(k + 1)^p \equiv k + 1 \ (\text{mod } p)$

Setting a to $(k + 1)$ gives the statement of the Fermat's little theorem.

$a^p \equiv a \ (\text{mod } p)$

The following are alternative statements:

$a^{p-1} \equiv 1 \pmod{p}$

$a^{p-1} - 1 \equiv 0 \pmod{p}$

Euler's Totient Function

Euler's Totient Function $\phi(n)$ counts the positive integers up to a given integer n that are relatively prime to n. In other words $\phi(n)$ is the number of integers k such that $1 \leq k \leq n$ for which $\gcd(n, k) = 1$.

For n = 9, $\phi(9)$ = 6

1, 2, 4, 5, 6, 7, 8 are relatively prime to 9 i.e gcd(a, n) = 1

3, 6, 9 are not, gcd(6,9) = 3

If m and n are relatively prime, then Totient function is multiplicative $\phi(mn) = \phi(m)\phi(n)$ i.e if $gcd(m, n)$ = 1 then $\phi(mn) = \phi(m)\phi(n)$

Euler's Theorem

Euler's theorem or Euler's Totient theorem generalizes Fermat's little theorem.

If a and n are coprime then $a^{\phi(n)} \equiv 1$ (mod n).

Eulers theorem can be used to reduce large powers modulo n.

Find 7^{222}(mod 10) i.e. find ones place decimal digit.

7 and 10 are coprime, $\phi(10) = 4, 7^4 \equiv 1$ (mod 10).

7^{222} = $7^{(4*55+2)}$ = $(7^4)^{55}$ * $7^2(mod 10) = (1)^{55} * 7^2 === 49 === 9(mod 10)$

7^{222}(mod 10) = $(7^4)^{55}$ * 7^2 (mod 10) = $(1)^{55} * 7^2$(mod 10) = 9

If $a^{\phi(n)} = 1$ (mod n) then $(a^{\phi(n)})^k = 1$ (mod n) for any k.