Admin: PS2 due today, PS3 out

Project proposals due 3.24

Today:

- Repeated squaring
- Multiplicative inverses mod p
- Finding large primes
- GCD (Euclid's alg.)
- Order of elements
- Generators

- Next few weeks we will study public key cryptography. Most of public key cryptography uses finite groups (such as $\mathbb{Z}_n^*$, where n is either a prime or a product of two primes), and involves doing arithmetic operations over these groups.

- In the next lecture we will do a group theory review, and define the most common groups used in crypto.

- Today we will learn a few basic algorithms that we will use in order to generate these groups, do arithmetics over these groups, etc.

Repeated squaring

Goal: given an elemen a in a group (or field) and given a non-negative integer $b \in \mathbb{N}$, compute a^b efficiently.

- Trivial solution requires b multiplications
- Solution that requires $\leq 2 \cdot \log b$ multiplications:

$$a^b = \begin{cases} 1 & \text{if } b=0 \\ (a^{b/2})^2 & \text{if } b \text{ even} \\ a \cdot a^{b-1} & \text{if } b \text{ odd} \end{cases}$$

$\Rightarrow$ If b consists of k bits then this alg requires $\leq 2k$ multiplications (as opposed to 2^k)

$\approx$ few milliseconds for $a^b \pmod p$ p = 1024-bit integer

$\approx O(k^3)$ time for k-bit inputs.

Computing multiplicative inverses mod p:

Thm: [For GF(P) called "Fermat's Little Theorem"]

$$\forall a \in \underbrace{\mathbb{Z}_p^*}_{\{1,2,\ldots,p-1\}} \qquad a^{p-1} = 1 \pmod{p}$$

Corollary: $\forall a \in \mathbb{Z}_p^* \qquad a^p = a \pmod{p}$

Corollary: $\forall a \in \mathbb{Z}_p^* \qquad a^{-1} = a^{p-2} \pmod{p}$

$\Rightarrow$ Can use repeated squaring to compute inverses efficiently!

Example: $3^{-1} \pmod{7} =$

$3^5 \pmod{7} =$

$3 \cdot \underbrace{(\underbrace{3^2}_{2})^2}_{4} \pmod{7} =$

$5 \pmod{7}$

* How about computing inverses mod n where n not prime? (Stay tuned!)

Finding large primes

Q: How do we find a large k-bit prime number?

Generate & test: **do** $p \leftarrow$ random k-bit integer
until p is prime

- This method works because primes are "dense":

Prime Number Thm: $\approx 2^k / \ln(2^k)$ k bit primes

$\Rightarrow$ One of every $\approx 0.69k$ k-bit integers is prime.

Q: How do we test primality?

Proposal: Given p, check if $2^{p-1} \stackrel{?}{=} 1 \pmod{p}$

(Any composite p that satisfies this equation is called "base-2 pseudoprime")

This works w.h.p. for random p;
doesn't work for adversarially chosen p.

- Miller-Rabin primality test (randomized)
- [Agrawal-Kayal-Saxena 2002]: Deterministic primality test.

Greatest Common Divisor (GCD)

Divisors: $d \mid a \equiv$ "d divides a" (evenly, over $\mathbb{Z}$)

$\equiv \exists k$ s.t. $a = d \cdot k$

- d is a divisor of a if $d \geq 0$ & $d \mid a$.
  - $\forall d \in \mathbb{N}$ $d \mid 0$ $\quad (0 = d \cdot \underset{k}{\underset{\shortparallel}{0}})$
  - $\forall a \in \mathbb{N}$ $1 \mid a$ $\quad (a = 1 \cdot \underset{k}{\underset{\shortparallel}{a}})$
  - If $\underbrace{d \mid a}_{a = d k_1}$ & $\underbrace{a \mid b}_{b = a \cdot k_2}$ then $\underbrace{d \mid b}_{b = d \cdot (k_1 k_2)}$.

* The greatest common divisor of a & b is the largest of their common divisors.
  [but $\gcd(0,0) = 0$ by def]

Examples:

$\gcd(24, 30) = 6$

$\gcd(5, 0) = 5$

$\gcd(33, 12) = 3$

$\gcd(28, 15) = 1$

Def: a & b are relatively prime if $\gcd(a, b) = 1$.

Euclid's algorithm: for computing $\gcd(a,b)$ $[a,b \geq 0]$:

$$\gcd(a,b) = \begin{cases} a & \text{if } b=0 \\ \gcd(b, a \bmod b) & \text{else} \end{cases}$$

Example:

$$\begin{aligned} &\gcd(7,5) \\ &= \gcd(5,2) \\ &= \gcd(2,1) \\ &= \gcd(1,0) \\ &= 1. \end{aligned}$$

- Running time $\propto \log(a) \cdot \log(b)$ bit operations.

  (polynomial running time: a shink by factor of 2 every two iterations.)

Thm: $\forall a,b \in \mathbb{Z}$ $\exists x,y \in \mathbb{Z}$ s.t. $ax+by = \gcd(a,b)$.

x, y are "witnesses" for the gcd: $Q|a, Q|b \Rightarrow Q|ax+by$

Extended Euclid's alg: (Similar to Euclid's alg, but not only remainders are kept, but also quotients)

Example: $a=7$ $b=5$

eq(1) $7 = 7 \cdot 1 + 5 \cdot 0$

eq(2) $5 = 7 \cdot 0 + 5 \cdot 1$

eq(3) $2 = 7 \cdot 1 + 5 \cdot (-1)$ eq(1) − eq(2).

$$1 = 7\cdot(-2) + 5\cdot 3 \qquad eq(2) - 2\cdot eq(3)$$

Used in RSA

Computing modular multiplicative inverses with the Extended Euclid's Alg:

$\mathbb{Z}_n^* = \{a : gcd(a,n)=1\}$ = multiplicative group modulo n.

Suppose $a \in \mathbb{Z}_n^*$ (i.e., $1 \le a < n$ & $gcd(a,n)=1$)

How to comute $a^{-1} \pmod n$?

If n is prime: $a^{-1} = a^{n-2} \pmod n$

(using Fermat's Little Thm)

Otherwise:

Find x, y s.t. $a\cdot x + n \cdot y = 1$

so $a \cdot x = 1 \pmod n$

and hence $x = a^{-1} \pmod n$

Example: $5^{-1} = 3 \pmod 7$

So far: We can generate primes,

can multiply, exponentiate, find inverses in $\mathbb{Z}_p^*$ & $\mathbb{Z}_n^*$.

Order of elements (in $\mathbb{Z}_p^*$ or $\mathbb{Z}_n^*$):

Def: $\forall a \in \mathbb{Z}_n^*$

$$\text{order}_n(a) = \underline{\text{least}}\ t \in \mathbb{N} \text{ s.t. } a^t = 1 \pmod n$$

Recall: Fermat's Little Thm:

$$\forall \text{prime } p\ \forall a \in \mathbb{Z}_p^*\quad a^{p-1} = 1 \pmod p$$

For general n:

Euler's Thm $\forall n \in \mathbb{N}\ \forall a \in \mathbb{Z}_n^*\quad a^{\varphi(n)} = 1 \bmod n$

where $\varphi(n) = |\mathbb{Z}_n^*| = |\{a \in \{1, \dots, n-1\} : \gcd(a,n) = 1\}|$

Example: $\mathbb{Z}_{10}^* = \{1, 3, 7, 9\}$

$\varphi(10) = 4$

$3^4 = 1 \pmod{10}$

$\Longrightarrow$ $\varphi(n)$ is well defined $\forall n$, & $\text{order}_n(a)$ is well defined $\forall n\ \forall a \in \mathbb{Z}_n^*$.

What is the relationship between them?

<u>Def</u>: $\forall n \in \mathbb{N}$ $\forall a \in \mathbb{Z}_n^*$ $\langle a \rangle = \langle a \rangle_n = \{a^i \pmod n : i \geq 0\}$

= subgroup generated by a.

<u>Example</u>: $n=7$ $a=2$: $\langle a \rangle = \{1, 2, 4\}$

<u>Thm</u>: $\forall n \in \mathbb{N}$ $\text{order}_n(a) = |\langle a \rangle_n|$ (easy!)

<u>Thm</u>: $\forall$ prime p $\forall a \in \mathbb{Z}_p^*$ $\text{order}_p(a) \mid (p-1)$

<u>Thm</u> (generalization): $\forall n \in \mathbb{N}$ $\forall a \in \mathbb{Z}_n^*$

$$\text{order}_n(a) \mid \underset{|\mathbb{Z}_n^*|}{\overset{}{\underset{\parallel}{\varphi(n)}}}$$

Generators

Def: $\forall$ prime p, $\forall g \in \mathbb{Z}_p^*$, if $\text{order}_p(g) = p-1$
then g is a generator of $\mathbb{Z}_p^*$ (i.e. $\langle g \rangle = \mathbb{Z}_p^*$).

Thm: $\forall$ prime p $\forall$ generator $g \in \mathbb{Z}_p^*$ $\forall y \in \mathbb{Z}_p^*$

$\exists$ unique solution $x \in \{0, 1, \dots, p-1\}$ s.t. $g^x = y$

x is said to be the discrete logarithm of y, base g, modulo p.

Ex: $p = 7$, $g = 3$

$x =$	1	2	3	4	5	6
$g^x =$	3	2	6	4	5	1

Thm: $\mathbb{Z}_n^*$ has a generator (i.e., $\mathbb{Z}_n^*$ is cyclic)
iff n is $2, 4, p^m$ $2 \cdot p^m$ $m \in \mathbb{Z}$

Thm: $\forall$ prime p the number of generators in $\mathbb{Z}_p^*$
is $\varphi(p-1)$

$$= \left| \{ a \in \{1, \dots, p-1\} : \gcd(a, p-1) = 1 \} \right|$$

Example: $p = 11$ $\qquad |\{1,3,7,9\}|$

$$\mathbb{Z}_{11}^{*} \text{ has } \varphi(10) = 4 \text{ generators}$$

(They are 2, 6, 7, 8)

3 is not a generator since $3^5 = 3 \cdot \underbrace{3^2 \cdot 3^2}_{4} = 1 \mod 11.$

How to find a generator mod a prime p?

In general, seems to require knowledge of factorization of $p-1$.

While factoring is hard, we can create primes for which factoring $p-1$ is easy.

Def: If p & q are both primes & $p = 2q+1$ then p is called a safe prime and q is called a sophie Germain prime.

Examples:

$p = 23$ $\quad q = 11$

$p = 11$ $\quad q = 5$

$p = 59$ $\quad q = 29$

<u>Thm</u>: If $p = 2q+1$ is safe prime then $\forall a \in \mathbb{Z}_p^*$

$order_p(a) = \{1, 2, q, 2q\}$

* It is not hard to find safe primes

Empirically, the prob. that a prime p is safe $\approx \frac{1}{\ln(p)}$.

Can check if g is a generator in $\mathbb{Z}_{p=2q+1}$ easily:

Check: $g^2 \neq 1 \mod p$

$g^q \neq 1 \mod p$

$g^{p-1} = 1 \mod p$ ✓ By Fermat

We can use "generate & test" again:

For safe prime p:

do $g \leftarrow \mathbb{Z}_p^*$

until $order_p(g) = p-1$

Are generators common ?

<u>Thm</u>: If $p = 2q+1$ is a safe prime then

\# generators mod p

$= \varphi(p-1) = q-1$ (almost half of them!)

In general:

<u>Thm</u>: $\forall$ prime p # generators in $\mathbb{Z}_p^*$ $= \varphi(p-1)$

$$\geq \frac{p-1}{6 \ln\ln(p-1)}$$

So generate & test works well for finding generators modulo any prime p for which you know $\varphi(p-1)$