Today: Primality testing and factorization.

Next time: Game theory

Who cares? For RSA, you need to figure out if big numbers are prime.

To be sure RSA is secure, need a good understanding of how fast factorization can go.

Warm-up: is 123456789 prime?

no! checked divibility by 2, 3. it is divisible.

$$123456789 = 9 \cdot \underbrace{13717421}_{\text{how about that?}}$$

You could check all the possible factors: divide by everything up to $\sqrt{13717421}$.

If don't find a factor, it's prime.

This is "trial division". Pretty slow, but it works.

(it will find a factor of 13717421 eventually.)

<u>Better algorithm for primality testing.</u> (no help factoring!)

<u>Fermat's Little Theorem:</u>

$n^{p-1} \equiv 1 \bmod p$ (if n is not a multiple of p, p prime)

To check if 13717421 is prime:

Find $\underbrace{2^{13717420}}_{\text{really big.}}$ mod 13717421. If it's not 1, not prime!

really big.
but we have a fast algorithm!
"repeated squaring"

= 7682470

<u>not prime!</u>

Try: is 341 prime?

2^{340} mod 341

$(2^0 \equiv 1 \mod 341)$

$2^1 \equiv 2 \mod 341$

$2^2 \equiv 4 \mod 341$

$2^4 \equiv 16 \mod 341$

$2^8 \equiv 256 \mod 341$

...

$2^{256} \equiv 64 \mod 341$ (comp)

binary expansion

↓

$$2^{340} = 2^{256+64+16+4} = 2^{256} \cdot 2^{64} \cdot 2^{16} \cdot 2^{4} = \underline{\underline{1}}$$

↑ those we know

What does that tell us? It <u>might</u> be prime.

Try n=3!

~~By~~ $3^{340} \mod 341 = 56.$ <u>not prime!</u>

(34=11x31, I checked seperately.)

This test is very fast, but not 100%

reliable.

→ If $n^{p-1} \equiv 1 \mod p$, then p is probably prime.
(no guarantees)

→ If $n^{p-1} \not\equiv 1 \mod p$, definitely not prime.

Algorithm: to determine if m is prime.

- compute $2^{m-1} \mod m$.
  - → if $\neq 1$, ~~keep going~~ definitely not prime
  - → if $= 1$, try another base.

- keep going until you are bored. the more bases, the surer you are it's prime, but never 100% certainty.

DANGER: There are numbers that pass the test for every base but aren't prime: "Carmichael numbers".
Smallest one is $561 = 3 \times 11 \times 17$.

How fast is this?

- To check if n is prime using bases $2, 3, 5, 7, 11$:
  takes time $O(\log n)$. "about" big-Oh

- Compare to trial division: $O(\sqrt{n})$.

test is much faster, but very rarely gives false positives.

Factorization Pollard ρ algorithm.

Birthday problem: If 25 people are in a room, what's the chance that 2 of them have the same birthday?

$\mathbb{P}$(no two have the same birthday)

$$= \frac{365}{365} \cdot \frac{364}{365} \cdot \frac{363}{365} \cdot \ldots \cdot \frac{341}{365}$$

1st, 2nd, 3rd, 25th

$$= \frac{{}_{365}P_{25}}{365^{25}} = \frac{\binom{365}{25} \cdot 25!}{365^{25}}, \qquad {}_{365}P_{25} = \frac{365!}{340!}$$

$\overset{\text{calc.}}{=} 0.43$

$P\left(\begin{smallmatrix}\text{two people have the}\\ \text{same}\end{smallmatrix}\right) = 1 - \text{that} \approx 57\%$

To generalize this:

~~If we have~~

If we pick k random numbers from N options,

- What is the chance 2 are the same? $\left(\begin{smallmatrix}N=365\\ k=25\end{smallmatrix}\right)$ done

- If we fix N, how big does k need to be to guarantee at least a 50% chance that 2 are the same? $\left(\begin{smallmatrix}N=365\\ k=23 \text{ is enough}\end{smallmatrix}\right)$

$$1 - \frac{N}{N} \cdot \frac{N-1}{N} \cdot \ldots \cdot \frac{N+1-k}{N}$$

$$= 1 - \frac{\frac{N!}{(N-k)!}}{N^k} = 1 - \frac{N!}{N^k (N-k)!}$$

For second problem, we want the k that makes that > 50%.

if N=1000, when is $1 - \frac{1000!}{1000^k (1000-k)!} \approx 50\%$

Hard! One idea: we need

$$(1)\left(1-\frac{1}{N}\right)\left(1-\frac{2}{N}\right)\cdots\left(1-\frac{k-1}{N}\right) < \frac{1}{2}$$

$$\approx 1 - \frac{(1+2+\cdots+(k-1))}{N} + \left(\frac{1}{N^2}\text{ stuff}\right)$$

let's ignore.

so we need $\frac{1+2+\cdots+(k-1)}{N} > \frac{1}{2}$

so $\frac{\frac{k^2-k}{2}}{N} > \frac{1}{2}$ roughly $\frac{\frac{k^2}{2}}{N} > \frac{1}{2}$

$\frac{k^2}{N} > 1.$ so $\boxed{k > \sqrt{N}}$ this is right when N is really big!

Factoring

Suppose you want to factor N.

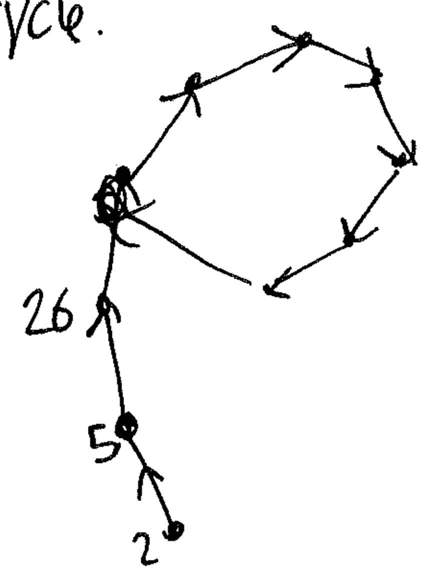
Here's what we do.

Set $X_0 = 2$ (or another favorite number)

$X_{i+1} = X_i^2 + 1 \mod N$.

this is basically a sequence of random numbers mod N.

eventually (probably after $\sim\sqrt{N}$ steps), you hit a number you already saw, and enter a cycle.



looks like "ρ", vaguely.

~~N=120~~

~~2, 5, 26, 77, 50, 101,~~

N=500

2, 5, 26, 177, 330, 401, 302, 205, 26, 177, 330, ...

repeat after 9 numbers!

Suppose N is a multiple of p. What happens to our sequence x_i if we look mod p instead of mod n?

It will probably repeat faster mod p than mod N!

$\sqrt{p}$ vs $\sqrt{N}$

eg N=500 is a multiple of 10!

repeats mod 10 faster than mod 500. ✓

Idea: if x_i and x_j agree mod p

(which is likely to happen pretty quickly),

then $x_i - x_j$ is a multiple of p.

To try to find a factor of N:

- Compute "random" sequence x_i
- Check $\gcd(N, x_i - x_j)$ for various i, j.

  Hope that finds a factor!