TU Graz

Prime testing

Daniel Perz

22. Januar 2020

History

Primes were studied from ancient times on.
Primality test were invented for numbers which where to large for trial division.
Since 1951 the largest primes were found by tests on computers.
Nowadays, primes play an important role in cryptography.

Trial division

Try whether $k \leq \sqrt{n}$ divides n.
Slow for large n.
Fast for small n.
Worth a try to check for some small k.

Fermats little theorem

Theorem

Let p be prime, a be an integer and $gcd(a, p) = 1$*, then*

$$a^{p-1} \equiv 1 mod\ p.$$

Problem: there are infinitely many composite numbers n, such that

$$a^{n-1} \equiv 1 mod\ n$$

holds for every integer a with $gcd(a, n) = 1$.

Strong Fermat test

Theorem

Let $n = d2^r + 1$ and d odd. Then for every a one of

$$a^d \equiv 1 \mod n$$
$$a^{d2^r} \equiv -1 \mod n$$

holds if and only if n is prime.

Generalised Fermat's little theorem

Theorem

For every integer a the equation

$$(X + a)^{n-1} \equiv X^{n-1} + a^{n-1} \mod n$$

holds if and only if n is prime.

Lucas test

Given $P > 0$, Q.
Let $D = P^2 - 4Q$, $\delta(n) = n - \left(\frac{D}{n}\right)$.
Define $U_n(P, Q) = P \cdot U_{n-1}(P, Q) - Q \cdot U_{n-2}(P, Q)$ with $U_0(P, Q) = 0$, $U_1(P, Q)$.
If n is prime and $gcd(n, Q) = 1$ then

$$U_{\delta(n)} \equiv 0 mod\ n.$$

Baille-PSW

Baille-PSW is a combination of strong Fermat test and a Lucas test.
It is a pseudoprime test, no exceptions are currently known.
Algorithm, whether n is prime:

- Make a strong Fermat test with $a = 2$.
- Find D in $5, -7, 9, -11, 13, \ldots$ such that $\left(\frac{D}{n}\right) = -1$. Set $P = 1$ and $Q = \frac{1-D}{4}$.
- Make a Lucas test with parameters D, P and Q.

Pocklington test

Idea: If n is composite, then it has a factor $F \leq \sqrt{N}$.
Factor $N - 1 = AB$ such that $gcd(A, B) = 1$, the factors of A are known and $A > \sqrt{n}$.
n is prime if for each $p|A$, there is an integer a_p such that

$$a_p^{n-1} \equiv 1 mod\ n \tag{1}$$

$$gcd(a_p^{(n-1)/p} - 1, n) = 1. \tag{2}$$

APR-CL

Uses generalised Fermat's little theorem.
Algorithm and proof is quite advanced.
Used for primality tests for medium numbers.
Almost all inputs need polynomial time.

ECPP

ECPP is for elliptic curve primality proof.
Currently fastest algorithm for large numbers.
Produces a primality certificate.
Almost all inputs need polynomial time, but worst case time is not known.

AKS test

- Test whether $n = a^b$.
- Find the smallest r such that $o_r(n) > \log^2 n$.
- Check whether $1 < gcd(a, n) < n$ for some $a \leq r$.
- If $n \leq r$, n is prime.
- For all $1 \leq a \leq \lfloor \sqrt{\phi(r) \log n} \rfloor$ check if

$$(X + a)^n \equiv X^n + a \mod (X^r - 1, n) \tag{3}$$

- Output n is prime

AKS test

- Runs in polynomial time.
- Currently not used in practice.
- Proof only need basic arguments.

Testing with Sagemath

Sagemath uses the following steps to check whether a given n is prime:

- check whether n has a small divisor.
- use Baillie-PSW to check if n is composite.
- use Pocklington test if $n - 1$ is easily factored.
- use APRCL for medium sized n (less than 1500 bits).
- use ECPP for large n.

Runtime

Is $10^{80} + 129$ prime?
Pseudoprime: 169 μs.
Prime: 38.7 ms.

Summary

- Stronger versions of Fermat's little theorem.
- Applications to primality tests.
- Elliptic curves can be used for primality testing.
- Primality testing is possibly in polynomial time.
- Tests are also used in practice.