6.885 Algebra and Computation November 2, 2005

Lecture 15

Lecturer: Madhu Sudan *Scribe: Jinwoo Shin*

Although there were some randomized polynomial-time algorithms for testing primality, the first deterministic polynomial-time algorithm for testing primality was published in 2002 by Manindra Agrawal, Neeraj Kayal and Nitin Saxena. In the previous lecture, we understood its basic idea and studied some notions and concepts to understand its correctness. Today, we will finish the proof of its correctness.

1 Notations

For fixed n, define a nice r.

Definition 1 *r is nice for n if*
- $r \nmid n$ and r is a prime number
- $ord_r(n) > 4\log^2 n$ ($ord_r(n)$ is the smallest number k such that $n^k = 1 \pmod{\mathrm{r}}$)

Then a small nice r always exists.

Proposition 2 *There exist a prime number less than $32\log^6 n$ which is a divisor of n or nice for n.*

Proof Let τ be $[4\log^2 n]$, and N be $\prod_{i=0}^{\tau}(n^i - 1)$ $(\le n^{\tau^2})$. Suppose $ord_r(n) \le \tau$ and r is not a divisor of n. Then $r \mid n^i - 1$ for some $i \in \{1, ..., \tau\}$, so $r \mid N$.

$$\textit{The number of prime divisors of } N \le \log N \le \tau^2 \log n$$

If we try the first $k(= \tau^2 \log n + 1 \le 16\log^5 n + 1)$ primes as a candidate for r, one of them is nice or a divisor of n. From the well-known weak theorem about the number of primes(The k th prime is less than $2k\log k$),

$$\textit{The } k \textit{ th prime} < 2k\log k \le 2(16\log^5 n + 1)\log(16\log^5 n + 1) \le 32\log^6 n$$

■

Now define 'intorospectiveness', the most useful concept in our algorithm and its proof of correctness.

Definition 3 *For polynomial $f(x)$, the positive integer m, and ring R, we say that m is* ***intospective*** *for $f(X)$ if*

$$[f(X)]^m = f(X^m) \textit{ in } R$$

Then we can get the following facts as proved in the previous lecture.

Fact 4

1. If m is introspective for $f(X)$ and $g(X)$ in R, it is also introspective for $f(X) \cdot g(X)$ in R.
2. If m_1 and m_2 is introspective for $f(X)$ in R $(= \mathbb{Z}[X]/(n, X^r - 1))$, then so is $m_1 \cdot m_2$.
3. If m is introspective for $f(X)$ in R, then m is also introspective for $f(X)$ in R',the subring of R.

2 Preliminaries

In this section, introduce and summarize some constants and sets will be used in this note. (There are many variables in the proof of correctness of our algorithm. So I am supposed to summarize them before entering the main section.)

- n : our fixed number to test.
- r : r is nice for n and less than $32 \log^6 n$.
- p : the prime divisor of n (if exists) and greater than r.
- l : l can be any number between r and t.
- T, t : $T = \{n^i p^j | i, j \geq 0\}$, $t = |\{m \text{ (mod r)} | \mathrm{m} \in \mathrm{T}\}|$.
- S, S_t : $S = \{\prod_{a=1}^{l} (X+a)^{d_a} | d_a \geq 0\}$, $S_t = \{f \in S | deg\ f < t\}$.
- $h(X)$: $h(X)$ is an irreducible polynomial that divides $\frac{X^r - 1}{X - 1}$.
- R, K : $R = \mathbb{Z}[X]/(n, X^r - 1)$, $K = \mathbb{Z}[X]/(p, h(X))$.

We can check some properties of them which will play crucial roles in the proof of correctness of our algorithm.

Observation 5

1. $t < \ l \ < \ r \ < \ p$
2. $|S_t| = \binom{l+t-1}{l} \geq 2^t \ (l > t)$
3. If n is not a power of p, then $t > ord_r(n)$.
4. If n is introspective for $X + a$ $(\forall a \in \{1, ..., l\})$ in R, every element of T is introspective for every element of S in both R and K. (From the Fact 4)

The existences of T,t,S,S_t,K are dependent on the existence of p. So they will appear only in the proof of correctness, not the algorithm itself.

3 Algorithm

Our algorithm for testing primality of n like this.

```
TEST($n$) :
1. if $n = m^k$ for some integer $m$ and $k$, OUTPUT COMPOSITE.
2. if $\exists\ r \in \{1, ..., [32 \log^6 n]\}$ such that $r \mid n$, OUTPUT COMPOSITE
3. Find a nice $r$ for $n$ in $\{1, ..., [32 \log^6 n]\}$
4. (MAIN STEP) for $a = 1$ to $l$,
        $if\ (X + a)^n \neq X^n + a \text{ (mod n,X}^\text{r} - 1\text{)}$, OUTPUT COMPOSITE.
5. OUTPUT PRIME
```

The step 1, step 2 and step 3 run obviously in $O(poly(n))$. The step 4 also runs in $O(poly(n))$ because the size of r and $l(l < r)$ must be $O(poly(n))$ from step 3. So our algorithm runs in $O(poly(n))$. The remaining work is to prove correctness of our algorithm. The following theorem tells its correctness.

Theorem 6 *Suppose r is nice for n. If there is no factor of n less than r and n is introspective for $X + a$ ($\forall a \in \{1, ..., l\}$) in R($= \mathbb{Z}[X]/(n, X^r - 1)$), then n is a power of a prime p greater than r.*

Prove this theorem in the next section.

4 Correctness

To prove our theorem, suppose the following assumption for contradiction.

Assumption 7

1. r is nice for n
2. n is introspective for $X + a$ ($\forall a \in \{1, ..., l\}$) in R
3. n is not a power of p.

Remember the facts proved in the previous lecture.

1. There exist two distinct numbers m_1 and m_2 in T such that $m_1 = m_2$ (mod r) and $m_1, m_2 \leq n^{2\sqrt{t}}$.
2. Let $P(Y) = Y^{m_1} - Y^{m_2}$ in $K[Y]$, then every element in S is a root of P in K.

From these facts, we can conclude the following.

Fact 8 *There are at most $n^{2\sqrt{t}}$ elements of S in K.*

With this fact and the following lemma, we can prove our main theorem.

Lemma 9 *Under the Assumption 7, two distinct elements f and g in S_t are also distinct in K.*

Proof Suppose $f(X) = g(X)$ in K. Because f and g are distinct in $\mathbb{Z}_p[X]$ ($p > l$), $f(X) = g(X)$ (mod h(X)). Let $e(Y)$ be $f(Y) - g(Y)$ in K. Then the degree of e is less than t and X is a root of e. Because any m in T is introspective for f and g in K(from the Observation 5-4), $e(X^m) = f(X^m) - g(X^m) = f(X)^m - g(X)^m = 0$ in K, so X^m is a root of e. Now show the following claim.

Claim : If $m_1 \neq m_2$ (mod r), then $X^{m_1} \neq X^{m_2}$ (mod h(X))

If $h(X)$ divides $X^{m_1} - X^{m_2}$, $h(X)$ divides $X^{m_1 - m_2} - 1$(Suppose $m_1 > m_2$). Because $h(X)$ divides $X^r - 1$, $h(X)$ divides $X^{gcd(m_1 - m_2, r)} - 1$. If $m_1 \neq m_2$ (mod r), $X^{gcd(m_1 - m_2, r)} - 1 = X - 1$ and hence $h(X)|X - 1$. This contradiction shows the claim.

Because there are t distinct (in mod r) elements in T, the number of roots of $e(Y)$ is at least t. This is contradiction because the degree of e is less than t. ■

Prove our theorem which gives correctness of our algorithm.

Proof of the main theorem Suppose the Assumption 7 is true. Then the Lemma 9 and the Fact 8 tell $S_t \leq n^{2\sqrt{t}}$. Because $2^t \leq S_t$(from the Observation 5-2), it is sufficient to show $2^t > n^{2\sqrt{t}}$ for contradiction.

$$t > ord_r(n) \ (n \textit{ is not a power of } p)$$
$$\rightarrow t > ord_r(n) > 4\log^2 n \ (r \textit{ is nice for } n)$$
$$\rightarrow \sqrt{t} > 2\log n$$
$$\rightarrow 2^t > n^{2\sqrt{t}} \ \blacksquare$$