MATH 314 Spring 2018 - Class Notes

11/05/2018

Scribe: Braeden Dorsey

Summary: Today we analyzed the steps involved in the Prime Number Theorem as well as the Solovay-Strassen Primality Test and the Miller-Rabin Primality Test.

- In order to use the RSA system, we must be able to find appropriate prime numbers for p and q
- Not only must these numbers be random, but also large enough so that guessing them would be extremely difficult
- BUt how do we find these numbers?
- Firstly, we have the Prime Number Theorem, which gives us an idea of how many primes exist that are less than a number x.

Prime Number Theorem
A way to find the number of primes that exist <= x.
Represented by Pi(x)
Pi(x) = x/(ln x) + O(x/$(lnx)^2$) or about x/(ln x)
For example, the number of prime numbers less than 10 would be Pi(10) = 4.

- once you decide on a number, how do you make sure it is prime?
- there are two ways to go about testing a number's primality
- the first is called the Solovay-Strassen Primality Test.

Solovay-Strassen Primality Test for integer n
Step 1:
Pick a random integer **a** where $1 < a < n-1$
Step 2:
Compute the Jacobi Symbol for (a/n)
Step 3:
Compute $a^{(n-1)/2}$ (mod n)
Step 4:
If the 2 results are the same, then we say **n is "probably prime"**
If they are not the same, then we say **n is composite**
If **n isn't prime** then at least half of all possible a's result in composite

- The next way is the Miller-Rabin Primality Test

Miller-Rabin Primality Test for integer n

Step 1:

Pick a radnom integer **a** where $1 < a < n - 1$

Step 2:

Write $n - 1 = 2^k m$ where m is odd

Step 3:

Compute $b_0 = a^m$ (mod n)

If b_0 is **1 or -1** then return n is **probably prime**

If not, continue to next step

Step 4:

Compute $b_1 = b_0^2$ (mod n)

If b_1 is **-1** then return n is **probably prime**

If b_1 is **1** then return n is **composite**

Else, continue to next step

Step 5:

Compute $b_2 = b_1^2$ (mod n)

If b_2 is **-1** then return n is **probably prime**

If b_2 is **1** then return n is **composite**

Else, continue to next step

Step 5:

Using the rules for b_2, continue going until you reach b_{k-1}

If you never get a **-1**, then return n is **composite**

Example : n = 561

let a = 2

n-1 = 560 = 16*35 = 2^4*35 where m = 35

$b_0 = 2^{35}$ (mod 561) = 263 (mod 561) = **continue**

$b_1 = 263^2$ (mod 561) = 166 (mod 561) = **continue**

$b_2 = 166^2$ (mod 561) = 67 (mod 561) = **continue**

$b_0 = 67^2$ (mod 561) = 1 (mod 561) = **composite**

561 is **composite**

If **n is composite** then at **most 1/4** possible a's return **probably prime**

If we get **"probably prime" every time** then the probability that **n is actually composite is** $< 1/1024$