

Lecture 11: Prime Numbers And Discrete Logarithms

Lecture Notes on “Computer and Network Security”

by Avi Kak (kak@purdue.edu)

February 14, 2017

1:53pm

©2017 Avinash Kak, Purdue University



Goals:

- Primality Testing
- Fermat’s Little Theorem
- The Totient of a Number
- The Miller-Rabin Probabilistic Algorithm for Testing for Primality
- **Python and Perl Implementations for the Miller-Rabin Primality Test**
- The AKS Deterministic Algorithm for Testing for Primality
- Chinese Remainder Theorem for Modular Arithmetic with Large Composite Moduli
- Discrete Logarithms

CONTENTS

	<i>Section Title</i>	<i>Page</i>
11.1	Prime Numbers	3
11.2	Fermat's Little Theorem	5
11.3	Euler's Totient Function	12
11.4	Euler's Theorem	15
11.5	Miller-Rabin Algorithm for Primality Testing	18
11.5.1	Miller-Rabin Algorithm is Based on an Intuitive Decomposition of an Even Number into Odd and Even Parts	20
11.5.2	Miller-Rabin Algorithm Uses the Fact that $x^2 = 1$ Has No Non-Trivial Roots in Z_p	21
11.5.3	Miller-Rabin Algorithm: Two Special Conditions That Must Be Satisfied By a Prime	24
11.5.4	Consequences of the Success and Failure of One or Both Conditions	28
11.5.5	Python and Perl Implementations of the Miller-Rabin Algorithm	29
11.5.6	Miller-Rabin Algorithm: Liars and Witnesses	38
11.5.7	Computational Complexity of the Miller-Rabin Algorithm	40
11.6	The Agrawal-Kayal-Saxena (AKS) Algorithm for Primality Testing	43
11.6.1	Generalization of Fermat's Little Theorem to Polynomial Rings Over Finite Fields	45
11.6.2	The AKS Algorithm: The Computational Steps	50
11.6.3	Computational Complexity of the AKS Algorithm	52
11.7	The Chinese Remainder Theorem	53
11.7.1	A Demonstration of the Usefulness of CRT	57
11.8	Discrete Logarithms	60
11.9	Homework Problems	64

11.1: PRIME NUMBERS

- **Prime numbers are extremely important to computer security.** As you will see in the next lecture, public-key cryptography would not be possible without prime numbers.
- As stated in Lecture 12, an important concern in public-key cryptography is to test a randomly selected integer for its **primality**. That is, we first generate a random number and then try to figure out whether it is prime.
- An integer is prime if it has exactly two **distinct** divisors, the integer 1 and itself. That makes the integer 2 the **first prime**.
- We will also be very interested in two integers being **relatively prime** to each other. Such integers are also called **coprimes**. Two integers m and n are coprimes **if and only if their Greatest Common Divisor is equal to 1**. That is if $\gcd(m, n) = 1$. Therefore, whereas 4 and 9 are coprimes, 6 and 9 are not. [See [Lecture 5 for gcd.](#)]

- Much of the discussion in this lecture uses the notion of **co-primes**, as defined above. The same concept used in earlier lectures was referred to as **relatively prime**. **But as mentioned above, the two mean the same thing.**
- Obviously, the number 1 is **coprime** to every integer.

11.2: FERMAT'S LITTLE THEOREM

- Our main concern in this lecture is with testing a randomly generated integer for its primality. As you will see in Section 11.5, the test that is computationally efficient is based directly on Fermat's Little Theorem. [This theorem also plays an important role in the derivation of the famous RSA algorithm for public-key cryptography that is presented in Section 12.2.3 of Lecture 12. Yet another application of this theorem will be in the speedup of the modular exponentiation algorithm that is presented in Section 12.5 of Lecture 12.]
- The theorem states that if p is a **prime number**, then for **every integer** a the following must be true

$$a^p \equiv a \pmod{p} \quad (1)$$

Another way of saying the same thing is that for any prime p and any integer a , $a^p - a$ will always be divisible by p . [Review the notation of modular arithmetic in Lecture 5 to fully understand what this theorem is saying. As stated in that lecture, $a^p \equiv a \pmod{p}$ means that $a^p \bmod p = a \bmod p$. For example, $8^3 \equiv 8 \pmod{3}$ since $8^3 \bmod 3 = 2$ and, at the same time, $8 \bmod 3 = 2$.]

[Click here to download full PDF material](#)