MATH3815 project3 --- Diffie-Hellman Key Exchange simulation

implementing Fast Exponentiation and Miller-Rabin Primality Test

Designer: Dai Junyan

Email: daiju@kean.edu

Instructor: Dr. Pinata Winoto

Date: 2018/6/24

**Introduction:**

This program purposes on simulating Diffie-Hellman Key Exchange algorithm with Fast Exponentiation Modulus and Miller-Rabin Primality Test. The language applied in this program is JAVA.

**Methods:**

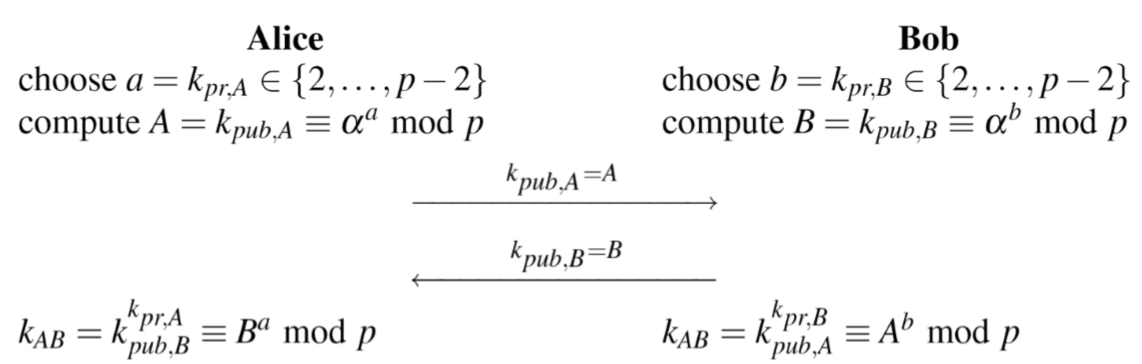
This program can be clearly divided into three parts: Diffie-Hellman Key Exchange, Fast Exponentiation Modulus and Miller-Rabin Primality Test. I will explain these three implementations one by one:

**Diffie-Hellman Key Exchange:**

Set-up: 1. The program asks user enter a prime number p. (the program will test if the entered p is likely a prime or not using Miller-Rabin Primality Test, it will be explained later)

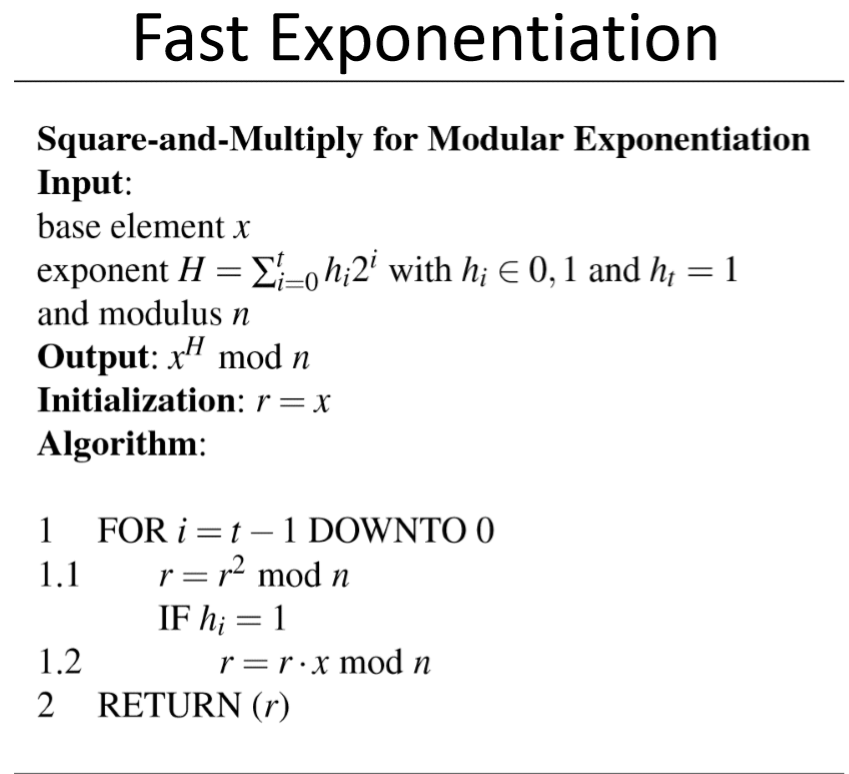
2. Randomly generate an integer g∈{2,3,…,p-2}

3. Publish p and g



**Fast Exponentiation Modulus:**

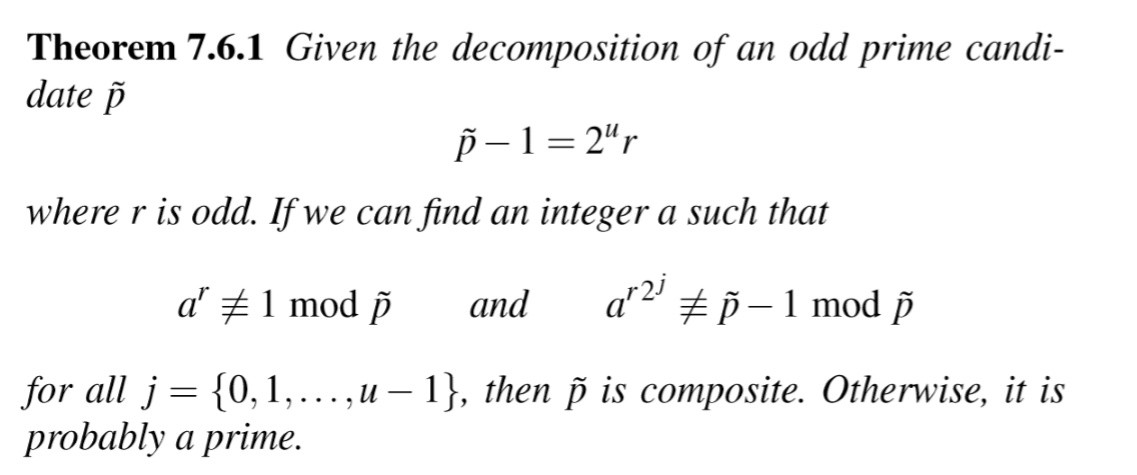
In the public and private computation and primality test, the Square-and-Multiply for Modular Exponentiation is applied to speed up the power modulus calculation. The rationale is explained in detail in textbook 7.4



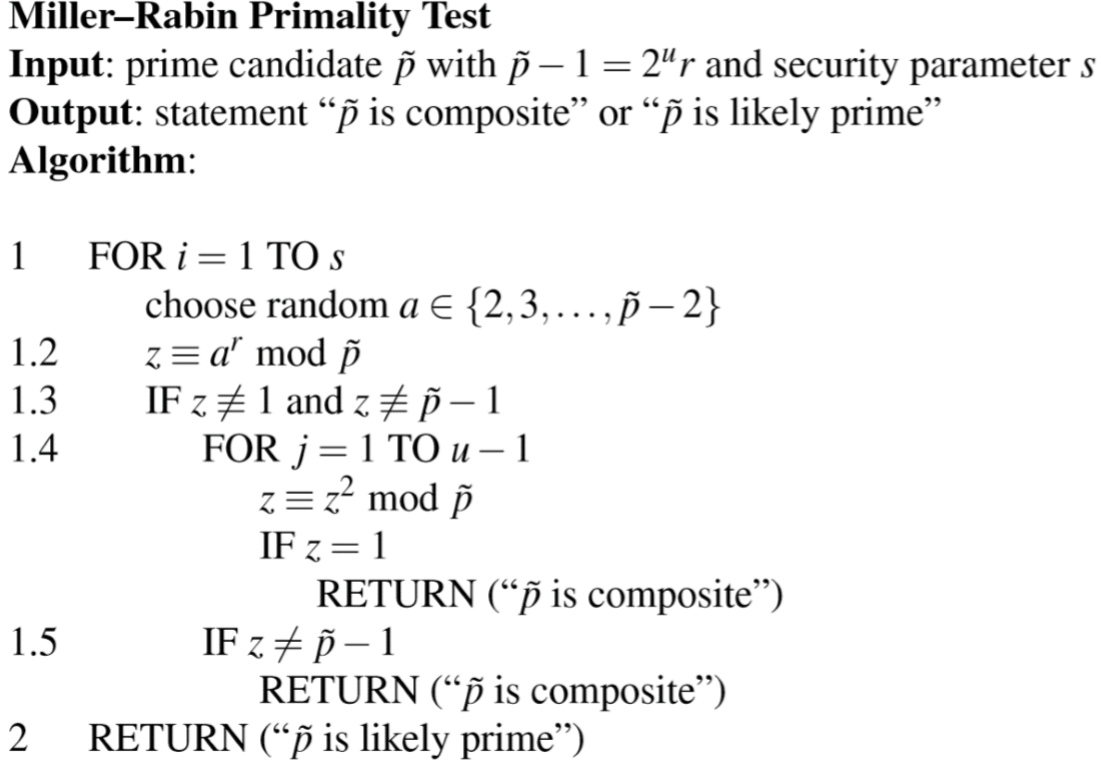
**Miller-Rabin Primality Test:**

Primality test can improve my program to randomly generate prime integers. But in this program, the user can directly enter a number and the program will tell user if the number entered is probably prime or certainly composite. There are two ways for primality test, Fermat Primality Test and Miller-Rabin Primality Test. Compare these two algorithm(textbook 7.6.2), Miller-Rabin Primality Test is relatively powerful method and is often used to generate RSA and DHKE primes.

**Miller-Rabin Theorem:**

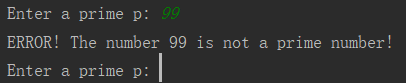


**Miller-Rabin pseudo code:**



**Output Demo:**

If I entered a composite integer at first, the error message would be prompted. And ask me to enter a new prime integer:



If I entered correctly(prime integer), the program starts Diffie-Hellman Key Exchange simulating:

