Uji Keprimaan Probalistik Solovay-Strassen dan Rabin-Miller

Maria Yus Trinity Irsan[1)], Loeky Haryanto [2)], Amir Kamal Amir[3)]
1), 2), 3)

Universitas Hasanuddin

Abstrak

Banyak skema kriptografi publik mengandalkan tingkat keamanannya pada kesulitan faktorisasi bilangan berukuran besar atas faktor-faktor prima yang seringkali terdiri atas ratusan digit. Jadi dibutuhkan suatu cara untuk menguji apakah suatu bilangan prima atau bukan prima secara efisien.

Uji Keprimaan Solovay-Strassen dan Uji Rabin-Miller adalah dua uji keprimaan secara probabilistik yang sangat efisien. Skripsi ini menguraikan latar belakang teori, prosedur dan cara penarikan kesimpulan kedua pengujian probabilistik tersebut. Dari uraian dan prosedur tes diperoleh hasil bahwa uji keprimaan Solovay-Strassen dirancang berdasarkan konsep simbol Jacobi sedangkan Rabin-Miller dirancang berdasarkan sifat-sifat akar dari bilangan bulat modulo bilangan yang diuji. Jika kesimpulan hasil uji terhadap sebuah bilangan bulat adalah bilangan tersebut mungkin prima, maka untuk uji Solovay-Strassen peluang kesimpulannya salah adalah lebih kecil atau sama dengan $\frac{1}{2}$, sedangkan untuk uji Rabin-Miller, peluang kesimpulannya salah adalah lebih kecil atau sama dengan $\frac{1}{4}$. Jadi jika uji dilakukan sebanyak mungkin, peluang mengambil kesimpulan yang salah bisa dibuat sekecil mungkin.

Kata Kunci: *algoritma solovay-strassen, algoritma rabin-miller, simbol legendre, simbol jacobi, teorema euler, teorema fermat, teorema sisa cina*

Abstract

Many public cryptography schemes have security that depend on the difficulties of factoring a large number into prime factors which could consist of hundred digits. Therefore, there is a need to have an efficient method to determine if a number is or is not a prime.

Several methods to test primality a number are called probabilistic test because there are only two conclusions of such methods: 1. The number is certainly composite; or 2. With error probability p, $0 < p < 1$; the number is probably prime.

Solovay-Strassen and Rabin-Miller primality tests are two efficient probability tests. This work studies the theories, the procedures and the derivations of the conclusions from the two tests. From the studies and procedures of the tests, the Solovay-Strassen test is designed based on the Jacobi symbol whereas the Rabin-Miller test is based on the properties of square root of number congruent modulo the tested number.

If the Solovay-Strassen test concludes that the tested number probably a prime then the probability that the conclusion incorrect is less than or equal to $\frac{1}{2}$, whereas for Rabin-Miller test, the probability is less than or equal to $\frac{1}{4}$. Therefore, if the test is iterated as many as possible, the error probability can be made as small as possible.

Keywords : *Chines Reminder Theorem, Euler's Theorem, Fermat's Theorem, Jacobi symbol, Legendre symbol, Rabin-Miller algorithm, Solovay-Strassen algorithm*

I. PENDAHULUAN

Masalah menghitung banyaknya waktu atau banyaknya langkah yang diperlukan untuk menyelesaikan suatu masalah merupakan bagian yang tidak dapat dipisahkan dari berbagai bidang aplikasi, sebagai contoh dalam berbagai algoritma untuk skema kriptografi, proses encoding dan decoding kode pengoreksi kesalahan, proses kompresi dan dekompresi data berbagai algoritma lainnya. Hal ini disebabkan karena dalam bidang aplikasi, masalah yang akan dipecahkan bisa berukuran sangat besar.

Skema kriptografi publik yang berbasis kesulitan faktorisasi, yang diawali oleh skema kriptografi RSA (1978), dan Rabin (1979) kemudian diikuti oleh puluhan skema kriptografi (juga skema tanda tangan digital) lain, mengandalkan tingkat keamanannya pada kesulitan (tepatnya *infeasibility*) mencari faktor-faktor prima dari suatu bilangan bulat positif n yang berukuran sangat besar dimana $n = pq$ dengan p dan q adalah bilangan prima. Dalam hal ini, diperlukan uji keprimaan (*primality testing*) untuk menentukan kedua bilangan prima p dan q yang juga berukuran sangat besar.

Skema kriptografi RSA misalnya menetapkan standar aman pada kesulitan faktorisasi bilangan $n = pq$ yang terdiri atas sekitar 1024 lebih bit atau sekitar 400 angka bilangan bulat n basis 10 dan masing-masing faktor prima p dan q kurang lebih berukuran sama, jadi salah satu diantaranya terdiri atas lebih dari 200 angka. Jika p adalah bilangan bulat prima yang terdiri atas 100 angka ($10^{99} \leq p \leq 10^{100}$) dan sebuah komputer super cepat dalam 1detik bisa menguji 100,000 bilangan bulat

Maria Yus Trinity Irsan
maria.triniy08@gmail.com

untuk menentukan secara berurutan bahwa semua bilangan bulat 3, 4, 5, ... $[\sqrt{p}]$ (terdiri atas lebih dari $10^{49} - 3$ bilangan bulat positif, karena $[\sqrt{p}] > 10^{49}$) bukan pembagi dari *p* maka waktu yang diperlukan oleh komputer tersebut untuk mendapatkan hasil bahwa *p* prima adalah lebih dari 3170979198.10^{27} tahun. Oleh sebab itu, sangat penting diketahui suatu uji keprimaan. Uji keprimaan yang paling banyak digunakan adalah uji Solovay-Strassen dan Rabin-Miller, karena keakuratannya dan ringan dalam komputasi. Dalam penulisan ini akan dibahas tentang konsep matematika pada kedua uji ini, kevalidan kedua uji, dan implementasinya pada pemrograman Maple.

II. METODA PENELITIAN

Penelitian ini lebih menekankan pada studi literatur walaupun dilengkapi dengan simulasi dengan menggunakan Maple. Jadi langkah awal dalam penelitian ini adalah studi kepustakaan yang meliputi:

i. studi teori bilangan secara umum,
ii. studi konsep simbol Legendre dan Jacobi yang menjadi dasar uji Solovay-Strassen,
iii. studi akar-akar dari 1 modolu *p*, dengan *p* prima, khususnya akar ke-*n* dari 1 modulu *p*, dimana *n* adalah bilangan yang diuji,
iv. implementasi langkah-langkah pengujian ke dalam Maple.

III. HASIL DAN DISKUSI

Pada bagian ini akan dibahas hasil dan pembahasan, namun demikian, terlebih dahulu disajikan definisi-definisi dan teorema-teorema pendukung sebagai berikut.

3.1 Landasan Teori

Teorema 3.1 (Teorema Sisa Cina) *Misalkan* $n_1, n_2, .., n_k$ *adalah bilangan bulat positif yang setiap pasangannya adalah koprima (FPB* $(n_i, n_j) = 1$, $i \neq j$*). Maka, untuk setiap bilangan bulat* $a_1, a_2, \dots, a_k$, *selalu ada bilangan bulat x yang merupakan penyelesaian dari system kongruensi simultan,*

$$x \equiv a_i (mod\ n_i)$$

Untuk i = 1, 2, …, k.

Prosedur menemukan nilai *x* adalah :

1. Hitung $N = n_1 n_2 \dots n_k$
2. Hitung $N_1 = \frac{N}{n_1}, N_2 = \frac{N}{n_2}, \dots, N_k = \frac{N}{n_k}$
3. Temukan invers perkalian $N_1^{-1}, N_2^{-1}, \dots, N_k^{-1}$ dengan menggunakan modulus $n_1, n_2, \dots, n_k$
4. Temukan *x* dengan menghitung:

$$x = (a_1 n_1 N_1^{-1} + a_2 n_2 N_2^{-1} + \cdots + a_k n_k N_k^{-1})$$

(Rifki Sadikin, 2012)

Teorema 3.2 (Akibat Teorema Sisa Cina) *Jika* $n_1, n_2, .., n_k$ *koprime, maka* $\mathbb{Z}_{n_1 . n_2 \dots n_k} \equiv \mathbb{Z}_{n_1} \times \mathbb{Z}_{n_2} \times \dots \times \mathbb{Z}_{n_k}$

Teorema 3.3 (Teorema Fermat) *Misalkan p adalah bilangan prima. Jika* $a \neq 0 (mod\ p)$, *maka* $a^{p-1} \equiv 1\ (mod\ p)$.

Definisi 3.5 (Fungsi Euler $(\boldsymbol{\phi(m)})$) *Bilangan* $\phi(m)$ *menyatakan banyaknya bilangan bulat positif x yang memenuhi:*

a. $1 \leq x < m$, *dan*
b. $(x, m) = 1$

Teorema 3.4 (Generalisasi Teorema Fermat oleh Euler) *Jika* $(x, m) = 1$ *maka* $a^{\phi(m)} \equiv 1\ (mod\ m)$.

Teorema 3.5 (*Euler's Totient Theorem)* *Jika n >1 dan a > 0, maka* $\phi(n) = n \prod_{p|n} (1 - \frac{1}{p})$, *untuk setiap bilangan prima p yang membagi n. Secara khusus, jika p prima maka* $\phi(p^a) = p^a \left(1 - \frac{1}{p}\right)$.

Definisi 3.6 (Residu Kuadratik) *Misalkan p adalah bilangan prima ganjil. Maka bilangan bulat x yang terletak pada interval* $1 \leq x \leq p - 1$ *disebut residu kuadratik jika terdapat y, sedemikian sehingga* $y^2 \equiv x\ mod\ p$ *mempunyai solusi. Bilangan tak nol lainnya disebut residu non kuadratik.*

Teorema 3.6 *Setengah dari anggota* $\mathbb{Z}_p^{\times}$ *adalah residu kuadratik dan* $QR_p = \{x \in \mathbb{Z}_p^{\times} \mid y^2 \equiv x\ (mod\ p), untuk\ suatu\ y \in \mathbb{Z}_p^{\times}\}$ *adalah subgrup dari* $\mathbb{Z}_p^{\times}$.

Definisi 3.7 Untuk setiap bilangan prima ganjil *p* dan bilangan bulat $a \geq 0$, didefinisikan *simbol Lengendre* $\left(\frac{a}{p}\right)$ sebagai berikut:

$$\left(\frac{a}{p}\right) = \begin{cases} 1, & \text{Jika } a \text{ mod } p \text{ residu kuadratik} \\ 0, & \text{Jika } a \text{ membagi } p \\ -1, & \text{Jika } a \text{ mod } p \text{ residu non kuadratik} \end{cases}$$

Teorema 3.7 (Teorema Lagrange) *Untuk Setiap grup berhingga G, jika H subgrup dari G, maka*

banyaknya anggota H (dinotasikan $|H|$) membagi banyaknya anggota G (dinotasikan $|G|$).

Teorema 3.8 *Untuk setiap bilangan prima ganji p dan bilangan bulat positif a,*

$$\left(\frac{a}{p}\right) = a^{\frac{p-1}{2}} \pmod p$$

Definisi 3.8 (Simbol Jacobi) *Misalkan n diuraikan sebagai perkalian bilangan – bilangan prima seperti berikut:*

$$n = p_1^{e_1} p_2^{e_2} p_3^{e_3} \dots p_k^{e_k}$$

maka,

$$\left(\frac{a}{n}\right) = \prod_{i=1}^{k} \left(\frac{a}{p_i}\right)^{e_i}$$

Sifat –Sifat Simbol Jacobi adalah sebagai berikut :

1. Jika $m \equiv 1 \bmod n$ maka,
$$\left(\frac{m}{n}\right) = 1$$
Jika $m \equiv 0 \bmod n$ maka,
$$\left(\frac{m}{n}\right) = 0$$
2. Jika $m_1 \equiv m_2 \bmod n$, maka
$$\left(\frac{m_1}{n}\right) = \left(\frac{m_2}{n}\right)$$
3. Perkalian symbol Jacobi:
$$\left(\frac{m_1 m_2}{n}\right) = \left(\frac{m_1}{n}\right)\left(\frac{m_2}{n}\right)$$
Khususnya jika $m = 2^k t$, t ganjil, maka
$$\left(\frac{m}{n}\right) = \left(\frac{2}{n}\right)^k \left(\frac{t}{n}\right)$$
4. $$\left(\frac{2}{n}\right) = \begin{cases} +1 & \text{jika } n \equiv \pm 1 (mod\ 8) \\ -1 & \text{jika } n \equiv \pm 3 (mod\ 8) \end{cases}$$
5. Misalkan m dan n adalah bilangan bulat ganjil. Maka
$$\left(\frac{m}{n}\right) = \left(\frac{n}{m}\right)$$
Kecuali untuk $n \equiv m \equiv 3\ (mod\ 4)$, untuk kasus ini $\left(\frac{m}{n}\right) = -\left(\frac{n}{m}\right)$

Definisi 3.9 (Bilangan Carmichael) *Misalkan m adalah bilangan majemuk. Jika $\forall a \in [2, m-2]$ yang memenuhi $(a, m) = 1$ berlaku $a^{m-1} \equiv 1 \bmod (m)$, maka m disebut bilangan Carmichael.* (Menezes,1997)

Teorema 3.9 (Teorema Little Fermat) *Jika $a^{n-1} \not\equiv 1 \bmod (n), a \not\equiv 0 \pmod n$, maka n adalah bilangan majemuk. Uji keprimaan Fermat adalah: Untuk menguji apakah n bilangan prima atau majemuk, pilih a secara acak kemudian hitung $a^{n-1} \pmod n$:*

i. Jika $a^{n-1} \equiv 1 \pmod n$, maka disimpulkan n prima.

ii. Jika $a^{n-1} \not\equiv 1 \pmod n$, maka diperoleh kesimpulan bahwa n majemuk.

Teorema 3.10 (Teorema Euler) *Jika $a^{\frac{(n-1)}{2}} \not\equiv \pm 1 \pmod n, \forall a \not\equiv 0 \pmod n$, maka n adalah bilangan majemuk. Uji Keprimaan Euler adalah: Pilih a secara acak dengan $a \not\equiv 0 \pmod n$, akan dihitung $a^{\frac{(n-1)}{2}} \pmod n$*

i. Jika $a^{\frac{(n-1)}{2}} \equiv \pm 1 \pmod n$, maka n disimpulkan prima.

ii. Jika $a^{\frac{(n-1)}{2}} \not\equiv \pm 1 \pmod n$, maka diperoleh kesimpulan bahwa n majemuk.

3.2 Algoritma Solovay-Strassen

Algoritma Solovay-Strassen didasari oleh kesamaan nilai simbol Jacobi dan teorema Euler, yaitu

$$\left(\frac{a}{n}\right) = a^{(n-1)/2} \bmod n$$

dengan *n* adalah bilangan yang akan di uji dan *a* adalah bilangan bulat acak yang terletak pada interval $1 \le a \le n-1$. Jika persamaan (1) terpenuhi maka disimpulkan *n* mungkin prima, jika tidak disimpulkan *n* majemuk. Berikut contoh sederhana untuk menggambarkan uji Solovay-Strassen.

Contoh 3.1 Akan diuji *n* = 71, dengan *a* = 35. Pertama-tama akan dicari nilai dari $\left(\frac{35}{71}\right)$ menggunakan sifat – sifat dari simbol Jacobi.

$$\left(\frac{35}{71}\right) = -\left(\frac{71}{35}\right) = -\left(\frac{1}{35}\right) = -1 \qquad (2a)$$

Selanjutnya, akan dihitung $a^{(n-1)/2} \bmod n$ (*a* = 35, *n* = 71)

$$35^{\frac{(71-1)}{2}} mod\ 71 = 35^{35} mod\ 71$$
$$= 70$$
$$\equiv -1\ mod\ 71 \qquad (2b)$$

Dari persamaan 2a dan 2b diperoleh,

$$\left(\frac{35}{71}\right) = 35^{(71-1)/2} mod\ 71 = -1$$

Jadi, disimpulkan *n* =71 mungkin prima.

Contoh 3.2 Akan diuji *n* = 87, dengan *a* = 50. Dengan cara yang sama dengan contoh 1, diperoleh

- $\left(\frac{50}{87}\right) = \left(\frac{2}{87}\right)\left(\frac{25}{87}\right)$

$= \left(\frac{25}{87}\right)$
$= \left(\frac{87}{25}\right)$
$= \left(\frac{12}{25}\right)$
$= \left(\frac{2}{25}\right)\left(\frac{6}{25}\right)$
$= \left(\frac{6}{25}\right)$
$= \left(\frac{2}{25}\right)\left(\frac{3}{25}\right)$
$= \left(\frac{25}{3}\right)$
$= \left(\frac{1}{3}\right)$
$= 1$ (3a)

- $50^{(87-1)/2} \; mod \; 87 = 50^{43} \; mod \; 87 = 8$ (3b)

Karena diperoleh $\left(\frac{50}{87}\right) \neq 50^{(87-1)/2} \; mod \; 87$, maka disimpulkan *n* = 87 adalah bilangan majemuk. Hal ini jelas, sebab $87 = 3 \times 29$.

Teorema berikut dirujuk dari [12], namun demikian, disini pemaparan bukti lebih lengkap.

Teorema 3.11 *Jika n adalah bilangan majemuk ganjil, dan* $a \in \{1, 2, \ldots, n-1\}$ *maka peluang algoritma Solovay-Strassen menyimpulkan n mungkin prima adalah* $\leq \frac{1}{2}$.
Bukti :

Misal $(n) = \left\{a | a \in \mathbb{Z}_n^{\times}, \left(\frac{a}{n}\right) \equiv a^{\frac{n-1}{2}} \; mod \; n\right\}$. Pertama akan ditunjukkan bahwa . $G(n)$ adalah subgrup $\mathbb{Z}_n^{\times}$.

- Jelas bahwa $G(n) \subseteq \mathbb{Z}_n^{\times}$.
- Akan ditunjukkan untuk setiap $a, b \in G(n)$ maka $ab \in G(n)$. Karena $a, b \in G(n)$, maka $\left(\frac{a}{n}\right) \equiv a^{\frac{n-1}{2}} \; mod \; n$ dan $\left(\frac{b}{n}\right) \equiv b^{\frac{n-1}{2}} \; mod \; n$. Dengan menggunakan sifat simbol Jacobi, maka diperoleh,

$$\left(\frac{ab}{n}\right) = \left(\frac{a}{n}\right)\left(\frac{b}{n}\right) \equiv a^{\frac{n-1}{2}} b^{\frac{n-1}{2}} \; mod \; n$$
$$= (ab)^{(n-1)/2} \; mod \; n$$

jadi, $ab \in G(n)$, artinya $G(n)$ tertutup terhadap operasi perkalian.

- Jelas bahwa 1 identitas $\mathbb{Z}_n^{\times}$ ada didalam $G(n)$, karena $\left(\frac{1}{n}\right) \equiv 1^{\frac{n-1}{2}} \; mod \; n$.
- Akan ditunjukkan untuk setiap $a \in G(n)$ terdapat $a^{-1} \in G(n)$ sedemikian sehinngga $aa^{-1} = 1$. Karena $a \in \mathbb{Z}_n^{\times}$, maka pasti ada $a^{-1} \in \mathbb{Z}_n^{\times}$, sekarang akan ditunjukkan $a^{-1} \in G(n)$. Dengan menggunakan sifat simbol Jacobi, diperoleh

$$\left(\frac{aa^{-1}}{n}\right) = \left(\frac{a}{n}\right)\left(\frac{a^{-1}}{n}\right)$$
$$\left(\frac{1}{n}\right) \equiv a^{\frac{n-1}{2}} \; mod \; n \left(\frac{a^{-1}}{n}\right)$$
$$1^{\frac{n-1}{2}} \; mod \; n \equiv a^{\frac{n-1}{2}} \; mod \; n \left(\frac{a^{-1}}{n}\right)$$

bagi kedua ruas dengan $a^{\frac{n-1}{2}}$

$$\left(\frac{1}{a}\right)^{\frac{n-1}{2}} \; mod \; n \equiv \left(\frac{a^{-1}}{n}\right)$$
$$(a^{-1})^{\frac{n-1}{2}} \; mod \; n \equiv \left(\frac{a^{-1}}{n}\right) \text{ atau}$$
$$\left(\frac{a^{-1}}{n}\right) \equiv (a^{-1})^{\frac{n-1}{2}} \; mod \; n$$

artinya, $a^{-1} \in G(n)$.

Dari keempat point diatas, terbukti $G(n)$ subgrup dari $\mathbb{Z}_n^{\times}$. Selanjutnya akan dibuktikan $|G(n)| \leq \frac{|\mathbb{Z}_n^{\times}|}{2}$. Dari teorema Lagrange (teorema 2.12) diketahui bahwa $|G(n)|$ membagi $|\mathbb{Z}_n^{\times}|$, jadi ada dua kemungkinan $|G(n)| = |\mathbb{Z}_n^{\times}|$ atau $|G(n)| < \frac{|\mathbb{Z}_n^{\times}|}{2}$. Akan ditunjukkan bahwa terdapat $a \in \mathbb{Z}_n^{\times}$, tetapi $a \notin G(n)$. Misalkan $n = p^k q$, dengan p prima ganjil, q bilangan bulat ganjil, $k \geq 2$, dan (*p*,*q*) = 1. Pilih $a = 1 + p^{k-1}q, a \in \mathbb{Z}_n^{\times}$. Sehingga diperoleh ,

$$\left(\frac{a}{n}\right) = \left(\frac{a}{p^k q}\right) = \left(\frac{a}{p}\right)^k \left(\frac{a}{q}\right) = 1.$$

Menggunakan teorema binomial diperoleh,

$$a^{\frac{n-1}{2}} = \left(1 + p^{k-1}q\right)^{\frac{n-1}{2}}$$
$$= \sum_{i=0}^{\frac{n-1}{2}} \binom{\frac{n-1}{2}}{i} \left(p^{k-1}q\right)^i$$
$$= 1 + \frac{n-1}{2} p^{k-1}q + \left(p^{k-1}q\right)^2 w$$
$$\equiv 1 + \frac{n-1}{2} p^{k-1}q \; (mod \; n).$$

Jika $\left(\frac{a}{n}\right) \equiv a^{\frac{n-1}{2}} \; (mod \; n)$, maka diperoleh,

$$1 \equiv 1 + \frac{n-1}{2} p^{k-1}q \; (mod \; n)$$
$$0 \equiv \frac{n-1}{2} p^{k-1}q \; (mod \; n)$$

artinya, $\frac{n-1}{2}p^{k-1}q$ adalah kelipatan n. Sehingga dapat ditulis,

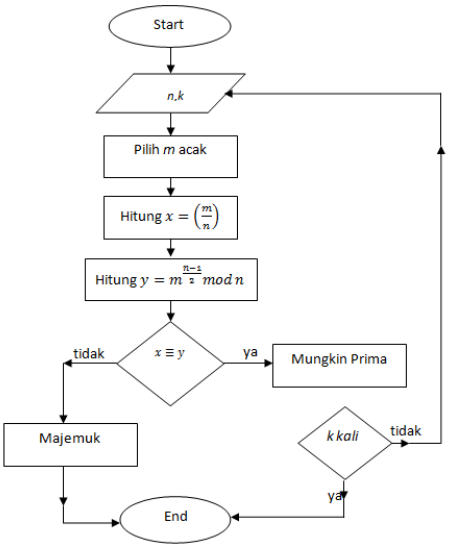
$$n|\frac{n-1}{2}p^{k-1}q \rightarrow p^k\, q|\frac{n-1}{2}p^{k-1}q \rightarrow p|\frac{n-1}{2}$$
$$\rightarrow p \equiv 1\ (mod\ n).$$

Tetapi hal ini kontradiksi dengan pernyataan awal bahwa $p \equiv 0\ (mod\ n)$. Jadi, $\left(\frac{a}{n}\right) \not\equiv a^{\frac{n-1}{2}}\ (mod\ n)$, sehingga $a \in \mathbb{Z}_n^{\times}$ tetapi $a \notin G(n)$, maka disimpulkan $|G(n)| \leq \frac{|\mathbb{Z}_n^{\times}|}{2}$. Jadi, probabilitas algoritma menyimpulkan kesimpulan yang salah adalah $\frac{|G(n)|}{|\mathbb{Z}_n^{\times}|} \leq \frac{\frac{|\mathbb{Z}_n^{\times}|}{2}}{|\mathbb{Z}_n^{\times}|} = \frac{|\mathbb{Z}_n^{\times}|}{2|\mathbb{Z}_n^{\times}|} = \frac{1}{2}$. ■

Dari Teorema 3.1 diatas telah terbukti bahwa peluang algoritma Solovay-Strassen menyimpulkan kesimpulan yang salah (tanpa perulangan) adalah lebih kecil atau sama dengan $\frac{1}{2}$, sehingga jika dilakukan perulangan sebanyak k kali misalnya (setiap perulangan dianggap saling lepas) maka error algoritma ini menjadi

$$\frac{1}{2}\cdot\frac{1}{2}\cdot \ldots \cdot\frac{1}{2} = \frac{1}{2}^k$$

Algoritma Solovay-Strassen dapat digambarkan dalam flowchart berikut:



Gambar 1. Flowchart Algoritma Solovay-Strassen

3.2.1 Implementasi Uji Solovay-Strassen (dalam Maple)

Berdasarkan flowchart algoritma Solovay-Strassen diatas, maka algoritma Solovay-Strassen dapat diimplementasikan ke dalam pemprograman Maple sebagai berikut:

Prosedur untuk mencari nilai dari $m^{\frac{n-1}{2}}\ mod\ n$

```
>
 eulerr := proc(n, m)
  local k;
  k := Power(m, (n - 1)/2) mod n;
  if type(k, n - 1) then return -1; end if;
  return k;
  end proc:
> euller(87, 50)
                        8
> euller(71, 35)
                       -1
```

Gambar 2. Prosedur mencari nilai $m^{\frac{n-1}{2}}$

Prosedur untuk mencari nilai $\left(\frac{m}{n}\right)$

```
jaccobian := proc(n, m)
 local nn, mm, sign;
 description "Pengujian Solovay Strassen";

 nn := n; mm := m;
 if mm = (1 mod n) or mm = (0 mod n) then
   return mm;
 end if;

 if mm > nn then
  return jaccobian(nn, mm mod nn);

 elif mm mod 2 = 0 then
  sign := 1;
  if nn mod 8 = 1 then
   sign := 1;
  elif nn mod 8 = 3 then
   sign := -1;
  elif nn mod 8 = 5 then
   sign := -1;
  elif nn mod 8 = 7 then
   sign := 1;
  end if;
  if mm = 2 then
   return sign;
  else
        mm
  mm := --; return sign·jaccobian(nn, mm);
        2
 end if;

elif mm mod 2 = 1 then
 if nn mod 4 = 3 and mm mod 4 = 3 then
  sign := -1;
 else
  sign := 1;
 end if;
   return sign·jaccobian(mm, nn);    end if;  end proc:
```

Gambar 3. Prosedur mecari nilai simbol Jacobi

Selanjutnya , kedua program diatas digabung/ akan dipanggil dalam satu program berikut sesuai dengan ketentuan uji Solovay-Strassen.

Prosedur program

```
>
solovaystrassen := proc(n, ulang)
local a, m, loop, mj;
a := rand(1 .. n − 1);

for loop by 1 from 1 to ulang do
    mj := 1;
    printf("Lolos Pengujian Ke %d\n", loop);
    m := a( );

    if jaccobian(n, m) ≠ eulerr(n, m) then mj := 0;end if;
    if type(mj, 0) then break;end if;
end do;
if mj = 0 then return "Majemuk";
end if;
return "Mungkin Prima";
end proc:
```

Gambar 4. Prosedur Algoritma Solovay-Strassen

Algoritma Rabin-Miller

Algoritma Rabin-Miller didasari oleh teorema Fermat yang menyatakan bahwa $a^{n-1} \equiv 1 \; mod \; n$ bila *n* adalah bilangan prima dan akar atau solusi *x* dari x^2 (mod *n*) yang memiliki paling sedikit empat akar jika *n* majemuk. Algoritma Rabin-Miller dapat dijelaskan sebagai berikut: 14actor bilangan integer positif yang akan diuji adalah bilangan ganjil *n* dengan $n \geq 3$. Maka *n* dapat ditulis,

$$n - 1 = m \times 2^k$$

dengan $k > 0$ (*k* adalah bilangan perpangkatan 2 terbesar yang habis membagi $n - 1$), dan *m* adalah bilangan ganjil. Langkah – langkah selanjutnya adalah sebagai berikut:

1. Ambil *a* acak yang terletak pada interval $1 < a \leq n - 1$. Hitung $T = a^m (mod \; n)$. Jika $T = \pm 1$ maka disimpulkan *n* mungkin prima.
2. Jika $T^2 = 1$ maka disimpulkan *n* majemuk dan algoritma berhenti. Jika $T^2 = -1$ maka disimpulan *n* mungkin prima. Jika $T^2 \neq \pm 1$ proses dilanjutkan ke langkah 3.
3. Hitung $T^{2^2} = a^{2^2 m} (mod \; n)$. Jika $T^{2^2} = 1$ maka disimpulkan *n* majemuk dan algoritma berhenti. Jika $T^{2^2} = -1$ maka disimpulan *n* mungkin prima. Jika $T^{2^2} \neq \pm 1$ proses (langkah 3) diulang hingga $T^{2^{k-1} m}$, jika sampai iterasi terakhir diperoleh $T^{2^{k-1}} > 1$ ($T \neq \pm 1$), disimpulkan *n* majemuk.

Contoh 3.3 Akan diuji *n* = 5937 dengan basis *a* = 5. Pertama dicari nilai *k* dan *m* sedemikian sehingga *n* – 1 = $2^k m$. Cara menemukan *k* dan *m* adalah dengan mempresentasikan 5937 – 1 = 5936 dalam bentuk biner, yaitu1011100110000, *k* adalah posisi 1 terkanan (indeks dimulai dari 0) yaitu pada indeks 4, jadi *k* = 4. Sedangkan nilai *m* adalah nilai biner dengan bit-bit 0 sebelah kanan indeks *k* dihapus sehingga *m* = $101110011_2 = 371_{10}$. Jadi, diperoleh *k* = 4 dan *m* = 371.

$$5937 - 1 = 2^4 . 371$$

Selanjutnya, hitung $T = a^m (mod \; n)$,

$$T = 5^{371} (mod \; 5937) = 1961$$

Karena bukan 1 atau -1 (5936), maka hitung $T = T^2$

$$T = T^2 = a^{2m} (mod \; n) = 5^{2.371} \; (mod \; 5937) = 4282$$

Karena bukan 1 atau -1, tidak dapat disimpulkan mungkin prima atau majemuk, maka dilakukan iterasi sampai $k - 2 = 4 - 2 = 2$.

Iterasi 1 : $T = 4282^2 \; (mod \; 5937) = 2068$

Iterasi 2 : $T = 2068^2 \; (mod \; 5937) = 1984$

Karena sampai iterasi $k - 2$ nilai *T* bukan 1 atau -1 maka dengan pasti disimpulkan *n* = 5937 adalah bilangan majemuk.

Lemma 3.1 *Misalkan* $n = p_1^{a_1} p_2^{a_2} \ldots p_k^{a_k}$, *dimana* $p_1, p_2, .., p_k$ *adalah bilangan prima berbeda. Jika* $x^m \equiv -1 \; mod \; n$, *untuk suatu bilangan bulat m, maka* $x^m \equiv -1 \; mod \; p$ *untuk setiap* $p|a$.

Bukti:

Dengan menggunakan Teorema Sisa Cina, dengan $n = p_1^{e_1} p_2^{e_2} \ldots p_k^{e_k}$ diperoleh,

$$x^m \equiv -1 \; mod \; n \equiv a_1 y_1 \frac{n}{p_1^{e_1}} + a_2 y_2 \frac{n}{p_2^{e_2}} + \cdots + a_k y_k \frac{n}{p_k^{e_k}}$$

$$1 \; mod \; n \equiv -a_1 y_1 \frac{n}{p_1^{e_1}} - a_2 y_2 \frac{n}{p_2^{e_2}} - \cdots - a_k y_k \frac{n}{p_k^{e_k}}$$

dengan $y_i = \left(\frac{n}{p_i^{e_i}}\right)^{-1} mod \; p_i^{e_i}$, sehingga $y_i \frac{n}{p_i^{e_i}} = 1 \; mod \; p_i^{e_i}$. Misalkan persamaan terakhir dikenakan modulo $p_1^{a_1}$ diperoleh,

$$1 \; mod \; n \equiv -a_1 . 1 \; mod \; p_1^{e_1} - 0 - \cdots - 0 = -a \; mod \; p_1^{e_1}$$

Jadi, $1 \; mod \; n \equiv -a_1 \; mod \; p_1^{e_1}$ atau $-a_1 \; mod \; p_1^{e_1} \equiv 1 \rightarrow -a_1 \equiv 1 \; mod \; p_1^{e_1} \rightarrow a_1 \equiv -1 \; mod \; p_1^{e_1}$. Demikian seterusnya untuk $p_2^{a_2}, p_3^{a_3}, \ldots, p_k^{a_k}$. Jadi, jika $x^m \equiv -1 \; mod \; n$ maka $x^m \equiv -1 \; mod \; p$ untuk setiap $p|n$. ■

Teorema berikut dirujuk dari [11], namun demikian, disini pemaparan bukti lebih lengkap.

Teorema 3.12 *Misal n bilangan bulat ganjil, n > 9,dan* $n = 1 + 2^k m$ *dengan* $k \geq 1$ *dan m ganjil. Jika*

$$B = \left\{ x \in \mathbb{Z}_n^{\times} | x^m = 1 \; atau \; x^{m 2^i} = -1, 0 \leq i < k \right\}$$

maka

$$\frac{|B|}{|\mathbb{Z}_n^{\times}|} \leq \frac{1}{4}$$

Bukti :

Misalkan, 2^l adalah perpangkatan 2 terbesar yang membagi $p-1$, untuk setiap p yang membagi n. Misalkan,

$$B' = \left\{x \in \mathbb{Z}_n^\times | x^{m2^{l-1}} = \pm 1\right\}.$$

Akan ditunjukkan $B \subset B'$. Jika $x \in B$ dengan $x^m = 1$, maka jelas $x \in B'$. Jika $x^{m2^i} \equiv -1 \ mod \ n$, dari lemma 3.1 $x^{m2^i} = -1 \ (mod \ p)$ untuk setiap p yang membagi n. Hal ini berarti untuk setiap p, perpangkatan 2 terkecil yang membagi order dari x mod p adalah 2^{i+1}. Juga 2^{i+1} membagi $p-1$ sebab order dari x (yaitu $m2^{i+1}$) di $Z_p^\times$ membagi order $Z_p^\times$ (yaitu $\varphi(p) = p-1$). Dari definisi l, $l \geq i+1$. Lebih jauh

$$x^{m2^{l-1}} = x^{m2^{l-1+i-i}} = \left(x^{m2^i}\right)^{2^{l-1-i}} = (-1)^{2^{l-1-i}}$$

bernilai 1 atau -1 tergantung pada apakah $l = I + 1$ atau $l > I + 1$. Jadi terbukti $B \subset B'$.

Dengan menggunakan Teorema Sisa Cina, banyaknya solusi $x \in \mathbb{Z}_n^\times$ dengan $x^{m2^{l-1}} = 1$ adalah sama dengan banyaknya perkalian solusi (akar) dari persamaan $x^{m2^{l-1}} = 1 \ mod \ p^{a_p}$ di mana p adalah bilangan prima dan p^{a_p} adalah perpangkatan terbesar dari p yang membagi n. Karena untuk setiap p, grup $\mathbb{Z}_{p^{a_p}}^\times$ adalah siklik dan himpunan akar-akar dari persamaan $x^{m2^{l-1}} = 1 \ mod \ p^{a_p}$ membentuk grup yang ordernya $m2^{l-1}$ maka banyaknya akar harus membagi $m2^{l-1}$ dan banyaknya akar dalam modulo p^{a_p} juga harus membagi $\varphi(p^{a_p}) = (1-p)p^{ap}$ (teorema 2.11), sehingga disimpulkan banyaknya akar dalam modulo p^{a_p} adalah

$$FPB\left((p-1)p^{a_p}, m2^{l-1}\right) = FPB((p-1,m)2^{l-1}$$

(p tidak membagi m).

Misalkan n adalah hasil kali sebanyak t perpangkatan 15actor-faktor prima yang berbeda. Menurut Teorema Sisa Cina, solusi dari

$$x^{m2^{l-1}} = 1 \bmod n$$

(yaitu sebuah 15actor $B' = \left\{x \in \mathbb{Z}_n^\times | x^{m2^{l-1}} = 1\right\}$) berasosiasi 1-1 dengan solusi dari t 15actor kongruensi

$$x^{m2^{l-1}} = 1 \bmod p^{a_p}.$$

Karena setiap 15actor kongruensi dalam modulo p^{a_p} memberikan $FPB((p-1,m)2^{l-1}$ solusi, jadi diperoleh,

$$\left|\left\{x \in \mathbb{Z}_n^\times | x^{m2^{l-1}} = 1\right\}\right| = \prod_{p|n} FPB\big((p-1),m\big)2^{l-1}.$$

Dengan cara yang sama, diperoleh banyaknya akar dari $x^{m2^l} = 1 \ mod \ p^{a_p}$ adalah FPB$(p-1,m)2^l$, juga banyaknya akar dari $x^{m2^{l-1}} = -1 \ mod \ p^{a_p}$ adalah $FPB((p-1,m)2^{l-1}$ atau dapat ditulis

$$\left|\left\{x \in \mathbb{Z}_n^\times | x^{m2^{l-1}} = -1\right\}\right| = \prod_{p|n} FPB\big((p-1),m\big)2^{l-1}$$

sehingga ,

$$|B'| = 2\left|\left\{x \in \mathbb{Z}_n^\times | x^{m2^{l-1}} = 1\right\}\right| = 2\prod_{p|n} FPB\big((p-1),m\big)2^{l-1}$$

dan,

$$\frac{|B'|}{|\mathbb{Z}_n^\times|} = 2\prod_{p|n} \frac{FPB\big((p-1),m\big)2^{l-1}}{(p-1)p^{a_p-1}}.$$

Misalkan $\frac{|B|}{|\mathbb{Z}_n^\times|} > \frac{1}{4}$, karena $B \subset B'$, maka $\frac{|B'|}{|\mathbb{Z}_n^\times|} > \frac{1}{4}$, persamaan diatas dapat ditulis

$$\frac{1}{4} < 2\prod_{p|n} \frac{FPB\big((p-1),m\big)2^{l-1}}{(p-1)p^{a_p-1}} \qquad (4)$$

Karena $FPB\big((p-1),m\big)2^{l-1}$ membagi $\frac{p-1}{2}$ maka ruas kanan dari pertidaksamaan (4) mempunyai nilaim paling besar 2^{1-t} dengan t adalah banyaknya p berbeda yang membagi n. Ini berarti $t \leq 2$.

Misalkan $t = 2$, jadi n memiliki dua pembagi yang berbeda. Misalkan salah satunnya adalah p, dan memiliki sifat p^2 membagi n maka $a_p \geq 2$, sehingga ruas kanan persamaan (4) memiliki nilai paling besar $\frac{2^{1-2}}{3} = \frac{1}{6}$, kontradiksi, maka $a^p = 1$. Jadi, $n = pq, p \neq q$. Persamaan (4) menjadi

$$\frac{1}{2^2} < 2\frac{FPB\big((p-1),m\big)2^{l-1}}{(p-1)p^{1-1}}\frac{FPB\big((q-1),m\big)2^{l-1}}{(q-1)q^{1-1}}$$

$$\frac{p-1}{FPB((p-1,m)2^{l-1}}\frac{q-1}{FPB((q-1,m)2^{l-1}}\frac{1}{2^2} < 2$$

$$\frac{p-1}{FPB((p-1,m)2^{l-1+1}}\frac{q-1}{FPB((q-1,m)2^{l-1+1}} < 2$$

$$\frac{p-1}{FPB((p-1,m)2^{l}}\frac{q-1}{FPB((q-1,m)2^{l}} < 2$$

Karena 15actor ruas kiri pertidaksamaan terakhir adalah bilangan bulat positif, maka keduanya adalah 1. Sehingga $p-1 = FPB((p-1),m)2^l$ dan $q-1 = FPB((q-1,m)2^l$. Berarti perpangkatan 2 yang membagi $p-1$ juga membagi $q-1$ adalah 2^l dan bagian ganjil dari $p-1$ dan $q-1$ membagi m. Berdasarkan hubungan $n = pq = 1 + 2^k m$ modulo bagian ganjil dari $p-1$ (yaitu $FPB((p-1),m)$), diperoleh

$$pq - q = 1 - q + 2^k m$$
$$q(p-1) = (1-q) + 2^k m$$
$$q(p-1) = -(q-1) + 2^k m$$

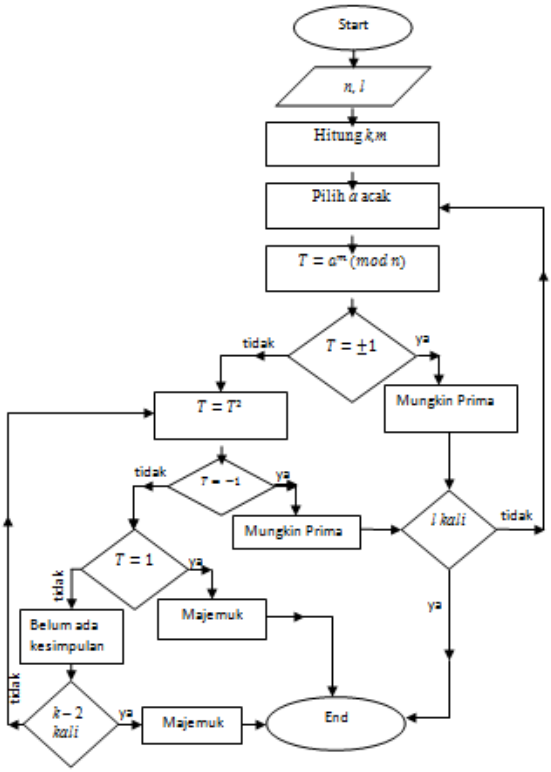
karena $q(p-1) \ mod \ FPB\big((p-1),m\big) = 0$, $2^k m \ mod \ FPB\big((p-1),m\big) = 0$, maka $(q-1) \ mod \ FPB\big((p-1),m\big) = 0$, sehingga $FPB\big((p-1),m\big)|(q-1)$. Sebaliknya, diperoleh $FPB\big((q-1),m\big)|(p-1)$. Jadi, $FPB\big((p-1),m\big) = FPB\big((q-1),m\big)$, menyebabkan $p-1 = q-1$. Kontradiksi dengan $p \neq q$. Oleh karena itu, $t = 1$. Jadi, $n = p^a$ dengan p bilangan prima ganjil dan

$a \geq 2$. Pertidaksamaan (4) menunjukkan $p^{a-1} < 4$, jadi p = 3, dan a = 2 menyebabkan n = 9 kontradiksi dengan hipotesis bahwa $n > 9$. Jadi pemisalan $\frac{|B|}{|\mathbb{Z}_n^{\times}|} > \frac{1}{4}$ harus diingkari. Disimpulkan $\frac{|B|}{|\mathbb{Z}_n^{\times}|} \leq \frac{1}{4}$. ■

Dari Teorema 3.2 diatas telah terbukti bahwa peluang algoritma Rabin-Miller menyimpulkan kesimpulan yang salah (tanpa perulangan) adalah lebih kecil atau sama dengan $\frac{1}{4}$, sehingga jika dilakukan perulangan sebanyak k kali (setiap perulangan dianggap saling lepas) maka error algoritma ini menjadi

$$\frac{1}{4} \cdot \frac{1}{4} \cdot \ldots \cdot \frac{1}{4} = \frac{1}{4}^{k}$$

Algoritma Rabin-Miller dapat digambarkan dalam bentuk flowchart sebagai berikut:



Gambar 5. Flowchart Algoritma Rabin-Miller

3.3.1 Implementasi Uji Rabin Miller (dalam Maple)

Algoritma pengujian bilangan prima Rabin-Miller dapat diimplementasikan ke dalam pemrograman Maple sebagai berikut

Prosedur Program

```
rabinmiller :=proc(n, ulang)
 local a, mp, T, m, k, biner, binerbaru, loop, b, eksloop;
 description "Pengujian Miller Rabin";

 k := 0;

 with(Bits);
 biner := String(n − 1, msbfirst);
 for loop by -1 from length(biner) to 1 do
  if type(biner[loop], "1") then
   break;
  end if;
  k := k + 1;
 end do;


  printf("K = %d\n", k)

  with(StringTools);
  binerbaru := SubString(biner, 1 .. length(biner) − k);
  m := convert(binerbaru, decimal, binary);
  printf("m = %d\n", m);
   a := rand(2 ..n − 1);

  for eksloop by 1 from 1 to ulang do
  mp := 0;
 b := a( );
  T := Power(b, m) mod n;

  printf("Perulangan Ke %d\n" , eksloop);
  if type(T, 1) or type(T, n − 1) then mp := 1;next;
  end if;

  for loop by 1 from 1 to k − 1 do
   T := T^2 mod n;
   printf("Perulangan dalam Ke %d T= %d\n" , loop, T);
   if type(T, 1) then
    return "Karena 1 Maka Majemuk";

   elif type(T, n − 1) then
    mp := 1; break;
    end if;
  end do;
  if mp ≠ 1 then return "karena sampai k-1 tdak diperoleh 1 atau -1, maka Majemuk"; end if;
   end do;
  if mp = 1 then return "Mungkin Prima";
  end if;
  #return "Majemuk";
  end proc:
```

Gambar 6. Prosedur Algoritma Rabin-Miller

IV. KESIMPULAN

Berdasarkan hasil dan pembahasan, maka disimpulkan beberapa hal, sebagai berikut :

1. Simbol Jacobi dan Teorema Euler merupakan konsep matematika yang diterapkan pada langkah awal algoritma Solovay-Strassen dan menjadi dasar dari algoritma ini.
2. Teorema Fermat dan pengujian akar kuadrat merupakan konsep matematika yang diterapkan dalam algoritma Rabin-Miller. Pengujian akar kuadrat yang dimaksudkan adalah menghitung apakah nilai-nilai
   $$a^m, a^{2m}, a^{4m}, a^{8m}, \ldots, a^{2^{k-1}m}$$
   sama dengan ± 1 dengan memperhatikan bahwa a^{im} adalah akar dari a^{2i}.
3. Tanpa perulangan, uji keprimaan menggunakan algoritma Solovay-Strassen memiliki probabilitas kesalahan lebih kecil atau sama dengan $\frac{1}{2}$, sedangkan menggunakan algoritma Rabin-Miller probabilitas kesalahannya lebih kecil atau

sama dengan $\frac{1}{4}$. Dengan melakukan perulangan, misalnya sampai k kali, probabilitas kesalahan uji keprimaan dengan algoritma Solovay-Strassen lebih kecil atau sama dengan $\left(\frac{1}{2}\right)^k$, sedangkan dengan algoritma Rabin-Miller probabilitas kesalahan kesalahannya lebih kecil atau sama dengan $\left(\frac{1}{4}\right)^k$. Sehingga disimpulkan, algoritma Rabin-Miller memiliki probabilitas ketepatan/kevalidan lebih tinggi dibandingkan dengan algoritma Solovay-Strassen. Tabel berikut adalah tabel perbandingan kevalidan kedua algoritma ini ditinjau dari banyaknya perulangan.

Banyak Perulangan	Probabilitas Kesalahan	
	Algoritma Solovay-Strassen	Algoritma Rabin-Miller
1	0,5	0,25
2	0,25	0,0625
3	0,125	0,015625
...	...	...
200	$6.223015278\,10^{-61}$	$3.872591915\,10^{-121}$
k	$\frac{1}{2}^k$	$\frac{1}{4}^k$

4. Algoritma Solovay-Strassen dan Rabin-Miller dapat diimplementasikan ke dalam pemograman Maple dengan pengujian sampai 200 digit.

REFERENSI

1. Menezes, P.Van Oorschot, and S.Vanstone. *Handbook of Applied Criptography*. CRC Press, 1997
2. Erawaty Nur,Haryanto Loeky; 2009.*Modul Pembelajaran Matakuliah Teori Bilangan*. Makassar: Universitas Hasanuddin
3. D.H. Lehmer. *On Euler's totient Function*. Math. Soc., 38 (1932), Hal: 745-751
4. D.J.Newmann. *Simple Analytic Proof of The Prime Number Theorem*. 1980. American Math.Monthly. Hal:693-696
5. *homepages.math.uic.edu/~marker/math435/rm.pdf*, diakses pada tanggal 7-10-2013
6. *http://www-math.ucdenver.edu/~wcherowi/courses/m5410/ctcprime.html*, diakses pada tanggal 9-11-2013
7. *home.sandiego.edu/~dhoffosc/teaching/cryptography/10-rabin-miller.pdf*, diakses pada tanggal 9-11-2013
8. Joe Hurd. Verification of the Miller-Rabin Probabilistic Primamility Test. *Journal of Logic and Algebraic Programming*, 50(1-2): 3-21, Mei-Agustus 2003. Special issue on Probabilistic Techniques for the Design and Analysis of Systems
9. Rosenberg Burt.1993 revised 2000. The Solovay-Strassen Primally Test
10. Sadikin Rifki. 2012. *Kriptografi Untuk Keamanan Jaringan*. Yogyakarta: AndiPublisher
11. Scoof Rene.2008.Four Primality testing algorithm. *Algorithmic Number Theory*.MSRI publication Vol.4,hal.102-104
12. [12] Scoof Rene.2008.Four Primality testing algorithm. *Algorithmic Number Theory*.MSRI publication Vol.4,hal.102-104