Algoritma Miller-Rabin untuk Uji Bilangan Prima

1 Pengantar Algoritma Probabilistik vs Deterministik

Algoritma deterministik selalu memberikan hasil yang sama untuk input yang sama, sedangkan algoritma probabilistik menggunakan nilai acak dalam prosesnya, yang memungkinkan hasil yang bervariasi tetapi dengan probabilitas kesalahan yang dapat dikontrol. Salah satu contoh algoritma probabilistik yang sering digunakan dalam teori bilangan adalah uji primalitas Miller-Rabin.

2 Deskripsi Masalah

Diberikan bilangan bulat n (n sangat besar), tentukan apakah n adalah bilangan prima atau bukan. Algoritma Miller-Rabin memberikan jawaban probabilistik dengan kemungkinan kesalahan yang kecil.

3 Penjelasan Algoritma Miller-Rabin

Algoritma Miller-Rabin adalah uji primalitas berbasis probabilitas yang bekerja dengan menguji apakah n lolos serangkaian pengujian modular berdasarkan bentuk faktorisasi $n-1$.

3.1 Langkah-Langkah Algoritma

1. Tulis $n-1$ dalam bentuk $2^s \cdot d$, dengan d ganjil.
2. Pilih bilangan acak a dalam rentang $[2, n-2]$.
3. Hitung $x = a^d \mod n$.
4. Jika $x = 1$ atau $x = n-1$, lanjutkan ke iterasi berikutnya.
5. Ulangi hingga $s-1$ kali: jika $x^2 \mod n = n-1$, lanjutkan ke iterasi berikutnya; jika tidak, n adalah komposit.
6. Jika semua percobaan gagal menunjukkan kompositas, maka n kemungkinan besar prima.

4 Probabilitas Kesalahan

Jika n adalah bilangan komposit, maka peluang algoritma gagal mendeteksi kompositas dalam satu uji adalah paling banyak $1/4$. Dengan melakukan k uji, probabilitas kesalahan menurun menjadi:

$$P(\text{kesalahan}) \leq \left(\frac{1}{4}\right)^k \tag{1}$$

Dengan memilih k yang cukup besar, probabilitas kesalahan dapat dibuat sangat kecil.

5 Contoh Perhitungan

Misalkan kita ingin menguji apakah $n = 561$ adalah bilangan prima.

1. Tulis $561 - 1 = 560 = 2^4 \cdot 35$.
2. Pilih $a = 2$, lalu hitung $x = 2^{35} \mod 561 = 263$.
3. Karena $x \neq 1$ dan $x \neq 560$, lanjutkan iterasi.

Algorithm 1 Algoritma Miller-Rabin

Require: Bilangan n, jumlah uji coba k
Ensure: Apakah n prima dengan probabilitas tinggi

if $n \leq 1$ **then**
    return False
end if
if $n = 2$ atau $n = 3$ **then**
    return True
end if
if n genap **then**
    return False
end if
Tulis $n - 1 = 2^s \cdot d$ dengan d ganjil
for $i = 1$ to k **do**
    Pilih bilangan acak $a \in [2, n-2]$
    Hitung $x = a^d \mod n$
    if $x = 1$ atau $x = n - 1$ **then**
        Lanjutkan ke iterasi berikutnya
    end if
    for $j = 1$ to $s - 1$ **do**
        Hitung $x = x^2 \mod n$
        if $x = n - 1$ **then**
            Lanjutkan ke iterasi berikutnya
        end if
    end for
    return False (Komposit)
end for
return True (Kemungkinan Prima)

4. Hitung $x^2 \mod 561$: $263^2 \mod 561 = 166$.
5. Hitung $166^2 \mod 561 = 67$.
6. Hitung $67^2 \mod 561 = 1$.
7. Karena tidak pernah mencapai 560, maka 561 adalah bilangan komposit.

6 Kompleksitas dan Perbandingan dengan Algoritma Deterministik

Algoritma deterministik seperti metode pembagian langsung memiliki kompleksitas $O(\sqrt{n})$, sedangkan Algoritma AKS memiliki kompleksitas $O(\text{polylog}(n))$. Namun, keduanya kurang efisien dalam praktik.

Sebaliknya, Miller-Rabin memiliki kompleksitas waktu $O(k \log^3 n)$ untuk k iterasi, yang jauh lebih cepat dan digunakan dalam banyak aplikasi dunia nyata, termasuk kriptografi. Dengan memilih k yang cukup besar, kemungkinan kesalahan dapat dibuat sangat kecil.