

Aplikasi Teori Bilangan pada Nomor Kartu Kredit

Kania Azrina - 13510058¹

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

¹13510058@std.stei.itb.ac.id

Abstract— Banyak sekali manfaat teori bilangan yang ada dalam kehidupan sehari-hari. Makalah ini akan membahas aplikasi teori bilangan dalam uji keabsahan kartu kredit menggunakan uji karakter. Teori bilangan adalah cabang dari matematika murni yang mempelajari bilangan bulat. Beberapa aplikasi teori bilangan dalam kehidupan sehari-hari adalah kriptografi, karakter uji dan fungsi hash. Kartu kredit adalah salah satu jenis kartu pembayaran yang memungkinkan penggunaannya untuk melakukan transaksi jual beli tanpa melakukan pembayaran pada saat itu juga. Nomor kartu kredit memiliki karakter uji yang dapat digunakan sebagai pengujian keabsahan nomor kartu kredit dengan menggunakan algoritma Luhn.

Index Terms— Teori bilangan, karakter uji, kartu kredit, algoritma Luhn

I. PENDAHULUAN

Sewaktu menimba ilmu di Sekolah Dasar ataupun Sekolah Menengah Pertama, banyak pelajar yang mempertanyakan apakah ilmu yang mereka pelajari ini akan berguna pada kehidupan sehari-hari atau hanya menjadi pengetahuan semata. Tetapi seiring beranjak dewasa, semakin terasa manfaat dari ilmu-ilmu tersebut, baik yang sudah disadari, maupun tidak. Dengan ilmu itulah, muncul banyak teknologi yang mempermudah peradaban manusia. Kesalahan-kesalahan yang terjadi, dapat dijadikan pembelajaran agar muncul teknologi baru yang dapat mengatasi kesalahan tersebut agar tidak terjadi kembali.

Teori bilangan adalah salah satu ilmu yang bermanfaat dalam kehidupan sehari-hari. Beberapa manfaat dari teori ini adalah keamanan dalam mengirimkan data, manajemen suatu data dalam memori komputer, dan lain-lain. Makalah ini akan membahas tentang salah satu aplikasi teori bilangan dalam kehidupan sehari-hari, yaitu uji keabsahan kartu kredit dengan menggunakan karakter uji, beserta algoritma dan cara pengujian yang dapat dicoba oleh siapa saja.

II. TEORI BILANGAN

A. Definisi Teori Bilangan

Teori bilangan (*number theory*) adalah bagian dari matematika murni yang khusus mempelajari bilangan

bulat. Para pakar teori bilangan atau yang disebut juga sebagai *number theorist* mempelajari bilangan prima (yang bila dikalikan, menghasilkan seluruh bilangan bulat) dan sifat-sifat dari bilangan yang dihasilkan dari bilangan bulat itu sendiri, seperti bilangan rasional.

Dulu, istilah teori bilangan lebih dikenal sebagai aritmatika. Lalu pada awal abad 20, namanya berubah menjadi teori bilangan. Aritmatika sendiri sering diartikan sebagai ‘perhitungan dasar’ pada masyarakat umum, sebagai ‘aritmatika Peano’ dalam logika matematika, dan sebagai ‘bilangan titik mengambang’ (*floating point arithmetic*) dalam ilmu komputer.

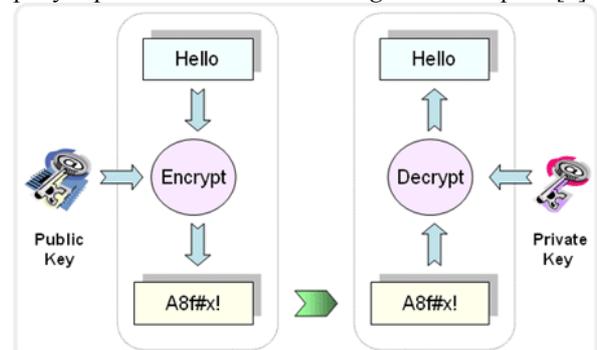
B. Aplikasi Teori Bilangan

1. Kriptografi

Kriptografi yang dalam Bahasa Yunani mempunyai arti “*Secret Writing*” merupakan ilmu dan seni untuk menjaga keamanan pesan dengan cara mengubahnya menjadi sebuah sandi atau bentuk lain yang tidak mempunyai makna bila dilihat sekilas. Tujuannya adalah agar pesan yang bersifat rahasia tidak dapat dibaca oleh pihak yang tidak mempunyai hak.

Ada dua text dalam kriptografi, yaitu plaintext dan ciphertext. Plaintext (pesan) merupakan sebuah data ada informasi yang dapat dibaca dan maknanya masih dapat dimengerti. Sedangkan ciphertext adalah pesan yang telah dijadikan sandi sehingga tidak mempunyai makna lagi. Untuk mengubah sebuah plaintext menjadi ciphertext, harus melalui proses bernama enkripsi (*encryption*). Dan proses untuk mengembalikannya bernama dekripsi (*decryption*).

Ada dua aplikasi untuk kriptografi, yaitu pengiriman data melalui saluran komunikasi dan aplikasi penyimpanan data di dalam *storage disk* komputer[1].



Gambar 1. Contoh aplikasi kriptografi pada kata “Hello”

2. Karakter Uji (*Check Digit*)

Karakter uji adalah bentuk pengecekan yang dilakukan untuk mendeteksi kesalahan. Angka pada karakter uji dihasilkan dari perhitungan angka-angka pada digit lainnya. Dengan karakter uji, seseorang dapat mendeteksi kesalahan-kesalahan kecil seperti kesalahan pengetikan [5]. Kesalahan pengetikan banyak macamnya. Berikut adalah kesalahan pada penulisan angka dan contohnya :

- Kesalahan satu karakter : 1->2
- Kesalahan transposisi: 12 -> 21
- Kesalahan kembar : 11 -> 22
- Kesalahan transposisi loncat : 132 -> 231
- Kesalahan kembar loncat: 131 -> 232
- Kesalahan fonetik : 60 -> 16 (dari 'sixty' menjadi 'sixteen')

Metode paling umum yang digunakan adalah menumlahkan jumlah semua angka pada digit dan membaginya dengan 10. Metode ini dapat mendeteksi kesalahan pada satu karakter, karena dapat mengubah nilai jumlah. Tetapi metode ini tidak dapat digunakan untuk mendeteksi kesalahan transposisi (pertukaran 2 digit), karena perubahan posisi bilangan tidak mengubah jumlahnya.

Algoritma-algoritma lain mempunyai metode yang lebih kompleks, seperti algoritma Luhn (1954) yang dapat mendeteksi 98% kesalahan transposisi satu karakter, 2% tidak dapat terdeteksi karena algoritma ini tidak dapat mendeteksi pertukaran antara 90 menjadi 09 dan algoritma Verhoeff (1969) yang dapat mendeteksi semua kesalahan pada satu karakter, Karakter uji terdapat pada Kode Universal Produk (*Universal Product Code*), ISBN 10, ISBN 13, Nomor Artikel Eropa (*European Article Number*), dan lain-lain.

3. Fungsi Hash

Data-data yang tersimpan di dalam memori komputer perlu ditempatkan pada suatu cara sehingga pencarian data tersebut dapat dilakukan dengan cepat. Setiap data (record) mempunyai field kunci yang unik yang membedakan antara satu record dengan record lainnya. Fungsi hash digunakan untuk menempatkan suatu record yang mempunyai k sebagai nilai kunci. Fungsi hash yang paling umum mempunyai bentuk

$$h(k) = k \text{ mod } m$$

m = jumlah alokasi memori

Fungsi h akan menempatkan record dengan kunci k pada suatu lokasi memori yang beralamat $h(k)$.

III. KARTU KREDIT

A. Definisi Kartu Kredit

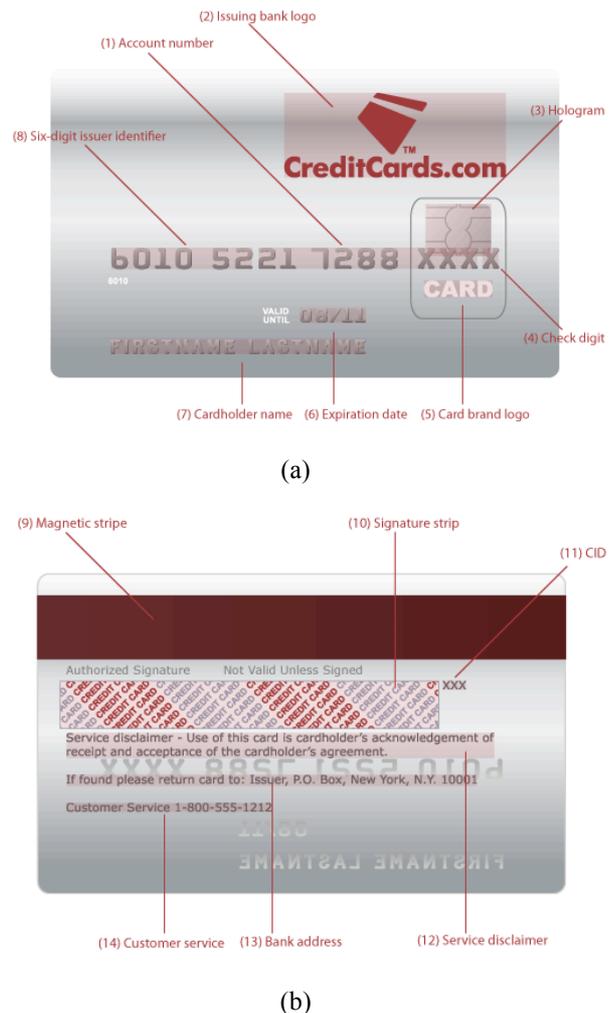
Kartu dibagi berdasarkan kategori penerbit kartu tersebut. Bisa berupa kartu bank (Visa, MasterCard, Discover), kartu petroleum (Sun Oil, Exxon), kartu hiburan dan pariwisata (America Express, Diners' Club), dan sebagainya. Kartu juga dapat terbagi sesuai cara

pembayarannya. Ada kartu kredit, kartu debit, dan lain-lain.

Kartu kredit adalah sebuah kartu yang dikeluarkan perusahaan yang dapat digunakan pemiliknya untuk membeli barang ataupun jasa tanpa harus membayar pada saat itu juga, melainkan pada saat yang telah ditentukan. Selain itu, pengguna kartu kredit dapat mencicil pembelian yang mereka lakukan selama jangka waktu tertentu, misalnya 6 bulan. Tetapi dengan menggunakan jasa cicilan ini, pengguna akan dikenakan bunga.

B. Anatomi Kartu Kredit

Susunan angka, garis magnetik, strip tanda tangan dan hologram mempunyai andil dalam setiap kartu kredit. Berikut adalah elemen-elemen yang ada pada kartu kredit.



Gambar 2. (a) bagian depan kartu kredit dan (b) bagian belakang kartu kredit

Bagian bernomor (1), (4) dan (8) akan dibahas pada bagian selanjutnya (3C).

(2) Logo penerbit kartu

Logo ini adalah logo lembaga yang meminjamkan uang yang dapat dipakai pengguna, bukan logo lembaga memroses pembayarannya.

(3) Hologram

Pada hologram ini tersimpan gambar tiga dimensi yang hanya dimiliki oleh kartu asli.

(5) Logo brand kartu

Ini adalah logo lembaga yang memproses pembayaran kartu. Lembaga pemroses pembayaran kartu terbesar adalah MasterCard, Visa, Discover dan American Express.

(6) Tanggal kadaluarsa

Tanggal kadaluarsa adalah batas tanggal keabsahan kartu kredit. Selain bulan dan tahun, terkadang ada tanggal awal atau tanggal akhir bulan tersebut.

(7) Nama pemilik kartu

Terdapat nama depan dan nama belakang pemilik kartu karena biasanya pada transaksi online, dibutuhkan nama yang persis sama dengan nama yang tertera pada kartu.

(9) Garis magnetik

Ada tiga warna strip magnet, yaitu hitam, coklat dan silver (seperti pada American Express). Ketiganya mempunyai 3 track, yaitu track 1, track 2, dan track 3. Track 3 tidak pernah digunakan secara langsung oleh 3 merek kartu utama seperti MasterCard, Diner's Club, Discover dan American Express dan biasanya tidak terlihat langsung pada kartu. Track 1 dan 2 biasanya dibaca oleh pembaca *point-of-sale*. Hanya sedikit informasi yang diperlukan untuk menyelesaikan transaksi yang ada pada track tersebut. Track 1 hanya satu-satunya track yang memiliki teks alfabet, dan hanya track 1 yang memiliki nama pemilik kartu kredit.

(10) Strip tanda tangan

Seperti tulisan yang ada di setiap strip tanda tangan, kartu kredit harus mempunyai tanda tangan pemiliknya agar sah. Tanda tangan ini digunakan untuk mengecek keabsahan tanda tangan saat melakukan transaksi.

(11) Nomor Identifikasi Kartu

CID (*Card Identification Number*) dalah 3 digit kode yang unik dari seluruh nomor yang ada pada kartu. Kode ini digunakan sebagai fitur pengaman tambahan.

(12) Layanan Penolakan

Pada umumnya, layanan penolakan (*service disclaimer*) menyatakan bahwa penggunaan kartu telah mengakui receipt dan penerimaan dari perjanjian antara penerbut kartu kredit dan pemilik kartu.

(13) Alamat Bank

Kartu kredit yang baik biasanya memiliki alamat pengembalian bila kartu hilang, penemu kartu dapat mengirimkannya kembali kepada bank penerbit.

(14) Layanan Konsumen

Terdapat nomor untuk layanan konsumen.

C. Nomor Kartu Kredit

Pada bagian bawah kartu kredit, terdapat nomor kartu

yang unik. Maksimal jumlah digit dari nomor kartu adalah 19 digit. Dalam nomor kartu ini, tersimpan informasi-informasi dasar tentang kartu tersebut.



Gambar 3. Contoh nomor kartu kredit

1. Identifikasi Lembaga Penerbit Umum atau *Major Industry Identifier* (MII)

MII berada pada digit ke-1. Digit pertama merepresentasikan kategori lembaga yang mengeluarkan kartu kredit tersebut. Berikut adalah daftar lembaga yang mengeluarkan kartu kredit dan nomor MII lembaga tersebut :

No. MII	Kategori Lembaga Penerbit
0	ISO/TC 68 dan industri lain
1	Penerbangan
2	Penerbangan dan industri lain
3	Pariwisata dan hiburan
4	Bank dan lembaga finansial lain
5	Bank dan lembaga finansial lain
6	Merchandizing dan bank
7	Petroleum
8	Telekomunikasi dan industri lain
9	Nasional

2. Identifikasi Lembaga Penerbit

Identifikasi ini terdapat pada digit ke-1 sampai 6, termasuk dengan nomor MII. Dengan kata lain, lembaga penerbit kartu kredit dapat berjumlah 10^6 atau 1 juta lembaga yang berbeda. Sebagai contoh, untuk kartu yang diterbitkan oleh Mastercard, 6 digit pertama adalah 51xxx-55xxx. Untuk kartu yang mempunyai nomor MII 9, 3 digit pertama adalah kode negara penerbit, sedangkan 2 digit setelahnya ditentukan oleh standar negara masing-masing.

3. Nomor Akun

Nomor akun pengguna berada pada digit ke-(7-(n-1)). Karena jumlah digit maksimal dari kartu kredit adalah 19, jumlah maksimal dari nomor akun pengguna adalah $19-7 = 12$. Maka jumlah maksimal pengguna kartu kredit adalah 10^{12} atau 10 triliun pengguna.

4. Karakter Uji

Pada digit terakhir setiap kartu kredit terdapat karakter uji untuk mengecek keasahan dari kartu kredit tersebut.

IV. UJI KEABSAHAN KARTU KREDIT MENGGUNAKAN TEORI BILANGAN

Sama halnya dengan kode ISBN yang menggunakan karakter uji sebagai metode pengecek keabsahan dari kode tersebut, karakter uji pada nomor kartu kredit juga menggunakan teori yang sama, tetapi dengan metode

pengujian yang berbeda.

A. Algoritma Luhn



Gambar 4. Hans Peter Luhn (1896-1964)

Algoritma Luhn adalah algoritma yang digunakan untuk keabsahan nomor-nomor identifikasi, seperti nomor kartu kredit yang dilihat dari karakter ujinya. Algoritma ini dibuat oleh Hans Peter Luhn (1896-1964), seorang ilmuwan komputer dari IBM. Algoritma ini biasanya digunakan untuk membedakan antara nomor kartu kredit yang sah dengan kartu kredit yang tidak, bukan digunakan untuk melindungi informasi rahasia kartu kredit dari pencurian data pribadi kartu.

Berikut adalah algoritma Luhn dalam bahasa Java.

```
//-----  
// Checks for valid credit card number using Luhn algorithm  
//-----  
  
public abstract class LuhnCheck {  
  
    //-----  
    // Filter out non-digit characters  
    //-----  
  
    private static String getDigitsOnly (String s) {  
        StringBuffer digitsOnly = new StringBuffer ();  
        char c;  
        for (int i = 0; i < s.length (); i++) {  
            c = s.charAt (i);  
            if (Character.isDigit (c)) {  
                digitsOnly.append (c);  
            }  
        }  
        return digitsOnly.toString ();  
    }  
  
    //-----  
    // Perform Luhn check  
    //-----  
  
    public static boolean isValid (String cardNumber) {  
        String digitsOnly = getDigitsOnly (cardNumber);  
        int sum = 0;  
        int digit = 0;  
        int addend = 0;  
        boolean timesTwo = false;  
  
        for (int i = digitsOnly.length () - 1; i >= 0; i--) {  
            digit = Integer.parseInt (digitsOnly.substring (i, i + 1));  
            if (timesTwo) {  
                addend = digit * 2;  
                if (addend > 9) {  
                    addend -= 9;  
                }  
            }  
            else {  
                addend = digit;  
            }  
            sum += addend;  
            timesTwo = !timesTwo;  
        }  
  
        int modulus = sum % 10;  
        return modulus == 0;  
    }  
}
```

```
//-----  
// Test  
//-----  
  
public static void main (String[] args) {  
    String cardNumber = "4408 0412 3456 7890";  
    boolean valid = LuhnCheck.isValid (cardNumber);  
    System.out.println (cardNumber + ": " + valid);  
    cardNumber = "4408 0412 3456 7893";  
    valid = LuhnCheck.isValid (cardNumber);  
    System.out.println (cardNumber + ": " + valid);  
    cardNumber = "4417 1234 5678 9112";  
    valid = LuhnCheck.isValid (cardNumber);  
    System.out.println (cardNumber + ": " + valid);  
    cardNumber = "4417 1234 5678 9113";  
    valid = LuhnCheck.isValid (cardNumber);  
    System.out.println (cardNumber + ": " + valid);  
}
```

Dalam algoritma di atas, langkah pertama yang dilakukan program tersebut adalah menyeleksi sebuah input string dan mengambil angka dalam string tersebut saja. Setelah terseleksi, dilakukan uji keabsahan. Proses pengalihan angka dalam digit bergantung pada jumlah angka tersebut. Angka pertama yang dikalikan dengan adalah angka (n-1), dengan n adalah jumlah digit. Dengan begitu, apabila jumlah digit bernilai genap, angka yang dilakukan adalah angka berdigit ganjil, begitu pula sebaliknya. Setelah dikalikan dengan dua, apabila hasilnya melebihi angka 9, secara otomatis program akan mengurangnya dengan angka 9 sehingga hasilnya akan berupa 1 digit saja. Setelah itu, semua angkanya dijumlahkan. Keabsahan dari angka-angka tersebut dilihat hasil penjumlahan tersebut, apabila hasilnya dapat dibagi dengan 10, maka angka- angka tersebut sah.

B. Aplikasi Pengujian

Seperti yang telah dijelaskan pada bagian 3C, nomor kartu kredit menyimpan beberapa informasi, sekaligus untuk mengecek keabsahan kartu kredit itu sendiri. Berikut adalah cara pengujiannya:

1. Untuk nomor kartu dengan jumlah digit genap, kalikan dua untuk setiap angka di pada digit bernomor ganjil, dan kurangi masing-masing angka tersebut dengan 9 apabila hasilnya lebih dari 9. Lalu jumlahkan semua angka tersebut. Apabila hasilnya tidak dapat dibagi dengan 10, maka nomor kartu kredit tersebut tidak valid.
2. Untuk nomor kartu dengan jumlah digit ganjil, lakukan hal yang sama angka pada digit bernomor genap.

Berikut adalah contoh pengujian pada nomor kartu kredit 4408041234567890 :

4	4	0	8	0	4	1	2	3	4	5	6	7	8	9	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Karena nomor kartu kredit diatas mempunyai 16 digit (genap), maka kalikan dengan 2 untuk seluruh angka pada berdigit ganjil.

8	4	0	8	0	4	2	2	6	4	10	6	14	8	18	0
---	---	---	---	---	---	---	---	---	---	----	---	----	---	----	---

Setelah itu, karena ada beberapa angka yang lebih dari angka 9, kurangi angka-angka tersebut dengan angka 9. Berikut adalah angka-angka tersebut setelah seluruhnya bernilai kurang dari 9.

8	4	0	8	0	4	2	2	6	4	1	6	5	8	9	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Setelah seluruh angka-angka tersebut dijumlahkan, akan

menghasilkan 67, yang bukan merupakan kelipatan 10, sehingga nomor kartu kredit tersebut tidak valid. Tetapi bila uji karakternya diubah dari 0 menjadi 3, akan menghasilkan 70 yang dapat dibagi dengan 10. Sehingga nomor kartu kredit menjadi sebuah nomor yang sah.

V. KESIMPULAN

Aplikasi teori bilangan dalam kartu kredit digunakan untuk menguji keabsahan kartu kredit. Tetapi, metode uji hanya dapat digunakan untuk mendeteksi kesalahan dalam penulisan / pengetikan nomor kartu kredit dan tidak dapat digunakan untuk mencegah pemakaian kartu kredit oleh pihak yang tidak berwenang.

REFERENSI

- [1] Munir, Rinaldi, *Matematika Diskrit Edisi 3*. Bandung: Palasari, 2007.
- [2] Anatomy of Credit Card Numbers, <http://www.merriampark.com/anatomycc.htm>. Diakses pada tanggal 9 Desember 2011, pukul 20.11 WIB.
- [3] Wikipedia, http://en.wikipedia.org/wiki/Credit_card. Diakses pada tanggal 9 Desember 2011, pukul 17.05 WIB.
- [4] Wikipedia, http://en.wikipedia.org/wiki/Number_theory. Diakses pada tanggal 10 Desember 2011, pukul 13.02 WIB.
- [5] Wikipedia, http://en.wikipedia.org/wiki/Check_digit.
- [6] Everything You Ever Wanted To Know About CC's, <http://euro.ecom.cmu.edu/resources/elibrary/everyc.htm>. Diakses pada tanggal 10 Desember 2011, pukul 18.00 WIB.
- [7] Tutorial Mengenal Aritmatika Modulo dalam Validasi Nomor Kartu Kredit – Part 1, <http://ajidotnet.wordpress.com/2009/12/15/tutorial-mengenal-aritmetika-modulo-dalam-validasi-nomor-kartu-kredit-part-1-basic-knowledge/>. Diakses pada tanggal 9 Desember 2011, pukul 16.43 WIB.
- [8] Anatomy of Credit Card, <http://knol.google.com/k/anatomy-of-a-credit-card#>. Diakses pada tanggal 10 Desember 2011, pukul 19.20 WIB.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 11 Desember 2011



Kania Azrina