

## ALGORITMA CHECK DIGIT

Djajasukma Tjahjadi

Sekolah Tinggi Manajemen Informatika dan Komputer LIKMI

Jl. Ir. H. Juanda 96 Bandung 40132

email: [djaja@likmi.ac.id](mailto:djaja@likmi.ac.id)

---

### ABSTRAK

Salah satu faktor yang menentukan kualitas informasi yang dihasilkan oleh suatu sistem adalah kualitas data masukan (input). Kualitas data masukan dapat ditingkatkan dengan menerapkan teknik validasi (*input validation routines*). Salah satu teknik validasi data masukan yang cukup populer adalah teknik *check digit*. Makalah ini ditujukan untuk mengulas 2 algoritma *check digit* yang cukup populer yaitu Algoritma Luhn dan Algoritma Verhoeff.

**Kata-kata kunci:** validation, Luhn, Verhoeff

---

### 1. PENDAHULUAN

Salah satu tujuan sistem informasi adalah menghasilkan informasi yang bermanfaat bagi pengambilan keputusan. Kualitas informasi yang dihasilkan pada dasarnya akan ditentukan oleh kualitas data masukan dan proses transformasi data menjadi informasi. Berdasarkan hal tersebut, maka para perancang sistem perlu menerapkan teknik-teknik untuk meningkatkan keakuratan data masukan. Ada beberapa teknik validasi untuk meningkat kualitas data masukan di antaranya (<http://http://www.hollyfield.kingston.sch.uk/gcseit/GCSE/validate.htm>):

- a. *Presence Check*, digunakan untuk mencegah agar data yang seharusnya diisi tidak dikosongkan.
- b. *Type Check*, digunakan agar data yang dimasukkan sesuai dengan tipe datanya, misalnya umur hanya bisa berupa angka.
- c. *Length Check*, digunakan agar data yang dimasukkan tidak melebihi jumlah karakter yang ditetapkan.

- d. *Range Check*, digunakan agar data yang dimasukkan berada pada batas minimal/maksimal yang ditetapkan.
- e. *Check Digit*, teknik menambahkan tambahan digit berupa angka untuk meningkatkan keakuratan data.
- f. *Parity Check*, teknik untuk memastikan bahwa data yang ditransmisikan dalam jaringan komputer dapat diterima secara akurat.

## 2. ALGORITMA CHECK DIGIT

*Check digit* adalah suatu teknik validasi dengan prinsip kerja menambahkan sebuah angka pada sebuah kode/nomor, misalnya nomor rekening nasabah, nomor kartu kredit dan sebagainya. Angka tambahan ini menjadi bagian yang tidak terpisahkan dari kode tersebut. Misalkan sebuah nomor rekening nasabah semula adalah 5786458. Dengan teknik ini, maka akan dihitung sebuah angka, yang disebut *check digit* dari kombinasi angka tersebut. Misalkan dari hasil perhitungan diperoleh bahwa *check digit*-nya adalah 5, maka nomor rekening nasabah menjadi 57864585. Pada saat nomor rekening nasabah diinput, maka komputer akan menghitung kembali *check digit* berdasarkan angka tersebut dan membandingkannya dengan hasil pengetikkan (yaitu angka 5 yang ikut diketik bersama-sama dengan nomor rekening tersebut). Jika *check digit* hasil perhitungan sama dengan *check digit* hasil pengetikkan maka dapat dipastikan bahwa nomor rekening yang diketik adalah valid.

Untuk menghitung angka tambahan (*check digit*) tersebut biasanya digunakan suatu algoritma perhitungan. Berdasarkan studi literatur yang dilakukan, setiap industri memiliki teknik penggunaan *check digit* yang berbeda-beda.

Ada dua algoritma perhitungan *check digit* yang cukup dikenal, yaitu:

- a. Hans P. Luhn, seorang ilmuwan IBM yang memperoleh hak paten atas temuan algoritma Luhn atau dikenal juga dengan algoritma Modulus 10 (Luhn, 1960)
- b. Jacobus Verhoeff, seorang matematikawan asal Belanda yang menciptakan Algoritma Verhoeff pada tahun 1969 (Verhoeff, 1969)

## 2.1 ALGORITMA LUHN

Algoritma Luhn ini telah digunakan di beberapa industri antara lain:

1. Perbankan, yaitu pada penomoran kartu kredit
2. Telekomunikasi, yaitu pada penomoran identitas perangkat bergerak secara internasional (International Mobile Equipment Identity/ IMEI) – (GSM Association, 2011)

Berikut ini adalah ilustrasi cara kerja algoritma Luhn:

Misalkan pada sebuah nomor kartu kredit 4563 7849 0347 5356, angka terakhir yaitu “6” sebenarnya merupakan *check digit* yang ditambahkan sebagai angka verifikasi. Untuk mengetahui apakah nomor kartu tersebut adalah nomor kartu yang valid atau tidak, maka dilakukan langkah sebagai berikut:

Langkah 1:

Kalikan setiap angka pada digit genap, dihitung dari kanan (belakang), yaitu sebagai berikut:

4	5	6	3	7	8	4	9	0	3	4	7	5	3	5	6
x2		x2		x2		x2		x2		x2		x2		x2	
8		12		14		8		0		8		10		10	

Langkah 2:

Jumlahkan seluruh digit angka, baik angka hasil perkalian maupun angka lainnya:

$$\text{Jumlah} = (8) + 5 + (1+2) + 3 + (1+4) + 8 + 8 + 9 + 0 + 3 + 8 + 7 + (1+0) + 3 + (1+0) + 6$$

Jumlah = 78, karena angka 78 bukan kelipatan 10 maka nomor kartu kredit di atas tidak valid.

## 2.2 ALGORITMA VERHOEFF

Algoritma Verhoeff memiliki tahapan yang lebih rumit, sehingga dapat mendeteksi jenis kesalahan yang lebih bervariasi dibandingkan Algoritma Luhn. Algoritma ini menggunakan beberapa bantuan tabel (matriks) sebagai berikut:

1. Tabel Dihedral Group, yaitu susunan angka yang simetris dalam hal ini digunakan D5, artinya susunannya memiliki bentuk yang simetris pada matriks 5 baris x 5 kolom, seperti ditampilkan pada Tabel 1

2. Tabel Permutasi (P) berdasarkan posisi (urutan dalam deretan angka=pos) dan angka ybs (num), seperti ditampilkan pada Tabel 2

Tabel 1 Tabel Dihedral (D)

D(i,j)		J									
		0	1	2	3	4	5	6	7	8	9
i	0	0	1	2	3	4	5	6	7	8	9
	1	1	2	3	4	0	6	7	8	9	5
	2	2	3	4	0	1	7	8	9	5	6
	3	3	4	0	1	2	8	9	5	6	7
	4	4	0	1	2	3	9	5	6	7	8
	5	5	9	8	7	6	0	4	3	2	1
	6	6	5	9	8	7	1	0	4	3	2
	7	7	6	5	9	8	2	1	0	4	3
	8	8	7	6	5	9	3	2	1	0	4
	9	9	8	7	6	5	4	3	2	1	0

Tabel 2 Tabel Permutasi (P)

P(pos, num)		num									
		0	1	2	3	4	5	6	7	8	9
pos	0	0	1	2	3	4	5	6	7	8	9
	1	1	5	7	6	2	8	3	0	9	4
	2	5	8	0	3	7	9	6	1	4	2
	3	8	9	1	6	0	4	3	5	2	7
	4	9	4	5	3	1	2	6	8	7	0
	5	4	2	8	6	5	7	3	9	0	1
	6	2	7	9	3	8	0	6	4	1	5
	7	7	0	4	6	9	1	3	2	5	8

Catatan: untuk pos=8 hasilnya sama dengan pos=0, pos=9 hasilnya sama dengan pos=1, dst.

Algoritma Verhoeff memiliki langkah sebagai berikut:

1. Dari deretan angka yang akan diproses buatlah array terdiri dari setiap digit angka dimulai dari paling kanan. Angka yang paling kanan memiliki posisi (pos) = 0 (nol)
2. Hitunglah P(pos, num) berdasarkan matriks permutasi P (lihat tabel 2)
3. Definisikan harga awal C (checksum) dengan angka Nol
4. Untuk digit angka pada langkah 1, hitunglah C yang baru yaitu hasil perhitungan Tabel Dihedral (D) berdasarkan rumus  $d(c, p(\text{pos}, \text{num}))$

Sebagai contoh dimisalkan sebuah deretan angka 1428570, di mana angka 0 di paling belakang merupakan cek digit. Apakah angka tersebut adalah angka yang valid berdasarkan algoritma cek digit Verhoeff ? Berikut ini adalah penerapan algoritma Verhoeff

Tahap 1: Buatlah array dengan mengambil setiap digit angka dari paling kanan

Hasil Tahap 1 = {0, 7, 5, 8, 2, 4, 1}

Tahap 2: Hitunglah P, C, dan d. Untuk memudahkan dibuat tabel sebagai berikut:

Pos	num	P (pos,num)	Nilai C sebelumnya	Nilai C yang baru =D(C , P(pos,num) )
0	0	0	0	0
1	7	0	0	0
2	5	9	0	9
3	8	2	9	7
4	2	5	7	2
5	4	5	2	7
6	1	7	7	0

Keterangan:

- Pos = posisi angka dimulai dari 0, 1, 2, dan seterusnya  
 Num = angka dalam setiap posisi. Jika pos = 0 maka ambillah angka terakhir dari deretan angka (dalam contoh berarti angka 0), dan seterusnya  
 P (Pos,num) = diperoleh dengan mencari ke Tabel Permutasi (tabel 2) berdasarkan baris (pos), dan kolom (num) yang sesuai. Contoh pada pos=1, num=7, maka carilah ke tabel 2 pada baris 1 kolom 7, sehingga diperoleh angka 0, maka  $P(1,7)=0$ . Dengan cara yang sama akan diperoleh  $P(2,5)=9$ , dan seterusnya  
 Nilai C  
 Sebelumnya = Untuk baris pertama nilai awal C didefinisikan 0, untuk baris selanjutnya nilai C sebelumnya adalah nilai C yang baru pada baris sebelumnya  
 Nilai C yang  
 baru = Dihitung dengan mencari ke Tabel Dihedral (Tabel 1) berdasarkan baris dan kolom yang sesuai. Contoh pada pos=2, num=5 diperoleh  $P(2,5) = 9$ , dan nilai C sebelumnya adalah 0, maka carilah ke tabel 1 pada baris 0, kolom 9 diperoleh angka 9, maka nilai C yang baru menjadi 9

Berdasarkan perhitungan pada tahap 2 diperoleh pada nilai C terakhir adalah 0, berarti cek digitnya sama dengan 0 (sama dengan angka nol yang paling kanan dalam deretan angka 1428570), sehingga dapat disimpulkan bahwa angka 1428570 merupakan angka yang valid menurut algoritma Verhoeff

### 3. KESIMPULAN

Berdasarkan studi literatur yang dilakukan dapat disimpulkan bahwa teknik *check digit* merupakan suatu teknik validasi data masukan yang efektif. Algoritma *check digit* yang paling populer adalah algoritma Luhn, yang telah banyak diaplikasikan pada berbagai industri misalnya penomoran kartu kredit, pemberian nomor seri perangkat lunak, nomor seri buku internasional, dan sebagainya.

Algoritma Verhoeff memiliki proses yang lebih rumit, namun dapat mendeteksi tipe kesalahan yang lebih beragam.

#### 4. DAFTAR PUSTAKA

- [1]. GSM Association. 2011. IMEI Allocation and Approval Guidelines Version 6.0 Luhn, H.P. 1960. *Computer for Verifying Number*. US Patent Office 2,950,048.
- [2]. Verhoeff, J. 1969. *Error Detecting Decimal Code*. Mathematical Centre Tract 29. Amsterdam
- [3]. <http://www.hollyfield.kingston.sch.uk/gcseit/GCSE/validate.htm>  
(diakses tanggal 15 April 2012 pk. 11.40)