

Aplikasi Modulo dalam Pengecekan Keaslian IMEI

Michael Hans 13518056¹
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
¹13518056@std.stei.itb.ac.id

Abstract—Teori bilangan memegang peranan penting dalam membantu pekerjaan manusia. Salah satunya adalah aritmatika modulo. Aritmatika modulo merupakan prinsip yang sering dipakai dalam melakukan validasi dari suatu nomor identifikasi (ID number). Salah satunya adalah dalam pengecekan keaslian IMEI. IMEI tersusun atas angka-angka yang mempunyai bagian dengan makna masing-masing dan satu buah angka Luhn untuk memvalidasi keaslian dari IMEI tersebut. Proses validasi tersebut memanfaatkan prinsip modulo yang mengacu pada formula Luhn atau algoritma modulo 10.

Keywords—modulo, bilangan bulat, luhn, valid.

I. PENDAHULUAN

Konsep mengenai bilangan dan operasi-operasi bilangan sudah dikenal sejak lama. Konsep bilangan ini sangat berguna dan dimanfaatkan untuk berbagai kepentingan, khususnya sebagai sarana dalam membantu pekerjaan manusia. Salah satunya adalah konsep mengenai teori bilangan. Teori bilangan memegang peranan penting dalam era industri saat ini. Bilangan prima memegang peranan penting secara dominan seperti dalam hal kriptografi dan hashing. Terkait dengan konsep aritmatika modulo, ternyata konsep modulo juga dimanfaatkan dalam kehidupan sehari-hari.

Salah satu penerapan nyata dari konsep modulo ini adalah terkait pengecekan validitas dari nomor ID atau sejenisnya. Tidak jarang kita menemukan berbagai jenis nomor ID di sekitar kita, seperti nomor ISBN, nomor kartu kredit, ID card, GSM, IMEI, dan sebagainya. Tentunya setiap nomor yang dikeluarkan merupakan nomor-nomor yang unik dan tentunya valid. Tidak jarang juga bilang sering terjadi pemalsuan ID atau nomor-nomor tertentu demi kepentingan komersial atau tindakan kejahatan lainnya. Oleh karena itu, pihak-pihak yang mengeluarkan nomor ID tersebut tentu tidak sembarang mengeluarkan suatu nomor begitu saja, melainkan terdapat identitas unik yang bisa divalidasi dengan cara tertentu, salah satunya adalah dengan menggunakan aritmatika modulo.

Pengecekan validitas yang akan dibahas dalam makalah ini adalah pengecekan keaslian suatu ponsel. Setiap handphone memiliki suatu identitas, yaitu nomor IMEI yang unik. Dalam makalah ini, salah satu aplikasi dari modulus adalah pengecekan keaslian dari suatu IMEI, dengan IMEI adalah salah satu nomor ID yang tertera pada suatu ponsel.

II. LANDASAN TEORI

A. Bilangan Bulat

Bilangan bulat adalah bilangan yang tidak mempunyai pecahan desimal, misalnya 5, 19, 23, -39, 1, 0, dan sebagainya. Bilangan bulat ini berlawanan dengan pengertian bilangan bulat yang adalah bilangan riil yang mempunyai titik desimal, seperti 5.0, 19.25, 0.02, dan sebagainya.

1. Sifat Pembagian pada Bilangan Bulat

Misalkan a dan b adalah dua buah bilangan bulat dengan syarat $a \neq 0$. Bilangan bulat a habis membagi b jika terdapat bilangan bulat c sedemikian sehingga $b = ac$.

Notasi : $a \mid b$ jika $b = ac$, $c \in \mathbb{Z}$ dan $a \neq 0$. Sebagai contoh, $4 \mid 12$ karena $12/4 = 3$ (bilangan bulat) atau $12 = 4 \times 3$. Tetapi 4 tidak habis membagi 13 karena $13/4 = 3.25$ (bukan bilangan bulat). Dengan kata lain, jika b dibagi dengan a , maka hasil pembagiannya berupa bilangan bulat. Pernyataan “ a habis membagi b ” bisa ditulis juga dalam bentuk “ b kelipatan a ” atau “ b habis dibagi a ”.

2. Teorema Euclidean

Misalkan m dan n adalah dua buah bilangan bulat dengan syarat $n > 0$. Jika m dibagi dengan n , maka terdapat dua buah bilangan bulat unik q (quotient) dan r (remainder), sedemikian sehingga $m = nq + r$ dengan $0 \leq r < n$. Bilangan n disebut pembagi (divisor), m disebut yang dibagi (dividend), q disebut hasil bagi (quotient), dan r disebut sisa (remainder). Notasi untuk mengekspresikan hasil bagi dan sisa dapat dinyatakan dalam bentuk mod dan div sebagai berikut.

$$q = m \text{ div } n$$

$$r = m \text{ mod } n$$

Contoh:

- a. 1987 jika dibagi dengan 97 memberikan hasil bagi 20 dan sisa 47. Pernyataan tersebut dapat dituliskan sebagai berikut.

$$1987 = 97 \cdot 20 + 47$$

$$1987 \text{ div } 97 = 20$$

$$1987 \text{ mod } 97 = 47$$

- b. Pernyataan -22 dibagi dengan 3 juga dapat dituliskan sebagai

$$-22 = 3 \cdot (-8) + 2$$

$$-22 \text{ div } 3 = -8$$

$$-22 \text{ mod } 3 = 2$$

3. Pembagi Bersama Terbesar

Dua buah bilangan bulat dapat memiliki faktor pembagi yang sama. Faktor pembagi bersama tersebut disebut sebagai faktor pembagi bersama terbesar (greatest common divisor -gcd) atau PBB. Misalnya 45 memiliki faktor pembagi 1, 3, 5, 9, 15, dan 45 sendiri, sedangkan 36 memiliki faktor pembagi 1, 2, 3, 4, 9, 12, 18, dan 36 sendiri. Faktor pembagi bersama dari 45 dan 36 adalah 1, 3, 9, yang terbesar adalah 9 sehingga disimpulkan PBB (45, 36) = 9. Definisi formal dari PBB dapat dinyatakan sebagai berikut.

Misalkan a dan b adalah dua buah bilangan bulat tidak nol. Pembagi bersama terbesar (PBB) dari a dan b adalah bilangan bulat terbesar d sedemikian sehingga $d \mid a$ dan $d \mid b$. Dalam hal ini, dapat dinyatakan bahwa PBB (a, b) = d .

Sifat-sifat dari pembagi bersama terbesar dapat dinyatakan sebagai berikut.

1. Misalkan a, b , dan c adalah bilangan bulat.
2. Jika c adalah PBB dari a dan b , maka $c \mid (a + b)$
3. Jika c adalah PBB dari a dan b , maka $c \mid (a - b)$
4. Jika $c \mid a$, maka $c \mid ab$

B. Aritmatika Modulo

1. Definisi

Aritmatika modulo adalah salah satu operasi aritmatika untuk memberikan sisa pembagian dari suatu pembagian a dengan b . Modulo dapat dinyatakan dalam operator berupa mod. Misalkan 23 dibagi 5 memberikan hasil = 4 dan sisa = 3 sehingga dapat dituliskan sebagai $23 \text{ mod } 5 = 4$. Definisi dari operator mod dinyatakan sebagai berikut.

Misalkan a adalah bilangan bulat dan m adalah bilangan bulat > 0 . Operasi $a \text{ mod } m$ (dibaca "a modulo m") memberikan sisa jika a dibagi dengan m . Dengan kata lain, $a \text{ mod } m = r$ sedemikian sehingga $a = mq + r$, dengan $0 \leq r < m$.

Bilangan m disebut modulus atau modulo, dan hasil aritmatika modulo m terletak di dalam himpunan $\{0, 1, 2, \dots, m - 1\}$. Hal ini terjadi karena sisa bagi dari suatu operasi pembagian tentu saja berada dalam rentang $0 - m-1$, dengan 0 menyatakan bahwa operasi pembagian telah habis membagi.

Berikut ini adalah contoh-contoh penggunaan modulo.

1. $23 \text{ mod } 5 = 3$ ($23 = 5 \cdot 4 + 3$)
2. $27 \text{ mod } 3 = 0$ ($27 = 3 \cdot 9 + 0$)
3. $6 \text{ mod } 8 = 6$ ($6 = 8 \cdot 0 + 6$)
4. $-41 \text{ mod } 9 = 4$ ($-41 = 9 \cdot (-5) + 4$)

2. Kongruen

Kongruen dari dua buah bilangan bulat a dan b menyatakan bahwa kedua bilangan bulat tersebut mempunyai sisa yang sama ketika dibagi dengan suatu bilangan bulat positif m . Hal ini disebut sebagai a dan b kongruen dalam modul m yang dilambangkan sebagai

$$a \equiv b \pmod{m}$$

Misalkan $38 \text{ mod } 5 = 3$ dan $13 \text{ mod } 5 = 3$, maka $38 \equiv 13 \pmod{5}$. Definisi formal dari kekongruenan dapat dinyatakan sebagai berikut:

Misalkan a dan b adalah bilangan bulat dan m adalah bilangan bulat > 0 , maka $a \equiv b \pmod{m}$ jika m habis membagi $a - b$.

Kekongruenan $a \equiv b \pmod{m}$ dapat pula dituliskan dalam hubungan

$$a = b + km$$

dengan k adalah sembarang bilangan bulat. Hal ini dapat dibuktikan dari definisi sebelumnya: $a \equiv b \pmod{m}$ jika $m \mid (a - b)$. Jika $m \mid (a - b)$, maka terdapat bilangan bulat k sedemikian sehingga $a - b = km$ atau $a = b + km$.

Berdasarkan definisi aritmatika modulo, pernyataan $a \text{ mod } m = r$ juga dapat dituliskan sebagai

$$a \equiv r \pmod{m}$$

3. Sifat Pengerjaan Aritmatika Modulo

Jika $a \equiv b \pmod{m}$ dan c adalah sembarang bilangan bulat maka

- (i) $(a + c) \equiv (b + c) \pmod{m}$
- (ii) $ac \equiv bc \pmod{m}$
- (iii) $a^p \equiv b^p \pmod{m}$ untuk suatu bilangan bulat tak negatif p

Jika $a \equiv b \pmod{m}$ dan $c \equiv d \pmod{m}$, maka

- (i) $(a + c) \equiv (b + d) \pmod{m}$
- (ii) $ac \equiv bd \pmod{m}$

C. Luhn Algorithm

Algoritma Luhn, Formula Luhn, atau biasa disebut *mod 10 algorithm*, adalah salah satu metode algoritma sederhana untuk mengecek kevalidan dari suatu *sequence* angka atau ID dengan metode modulo 10. Algoritma ini diciptakan oleh Hans Peter Luhn. Algoritma Luhn ini sudah banyak digunakan untuk berbagai pembuatan *user ID* atau *sequence* angka, seperti nomor kartu kredit, nomor IMEI, kartu ID penduduk Israel, nomor asuransi di Kanada, dan sebagainya.



Gambar 1. Hans Luhn Peter, pencipta Algoritma Luhn (Sumber : <https://researcher.watson.ibm.com/researcher>)

Berikut ini adalah algoritma dari Luhn Algorithm terkait pembuatan nomor identifikasi.

1. Lakukan pembacaan angka dari ujung kanan.
2. Setiap angka pada posisi ganjil tidak ada yang diubah. Sedangkan setiap angka pada posisi genap dikalikan dengan 2 sehingga diperoleh angka yang merupakan hasil perkalian dengan 2.
3. Lakukan penjumlahan digit dari masing-masing elemen dari larik sehingga diperoleh suatu sequence angka baru.
4. Tentukan nilai validasi sedemikian rupa sehingga penjumlahan sequence angka yang baru tersebut habis dibagi 10.

Misalkan diberikan suatu nomor akun dengan sequence angka berupa 7992739871x.

Nomor akun	7	9	9	2	7	3	9	8	7	1	x
Hasil kali posisi genap	7	18	9	4	7	6	9	16	7	2	x
Penjumlahan digit-digit	7	9	9	4	7	6	9	7	7	2	x

Penjumlahan dari seluruh digit pada baris ketiga adalah $67 + x$. Dalam kasus ini, x adalah *check digit*, yaitu digit yang akan digunakan untuk validasi nomor akun tersebut. Terdapat dua cara untuk mendapatkan *check digit* (x).

Cara pertama:

1. Jumlahkan seluruh digit-digit pada baris ketiga.
2. Kalikan dengan $k = 9$
3. Modulokan hasil sebelumnya dengan mod 10 sehingga diperoleh check digit.

Cara kedua:

1. Jumlahkan seluruh digit-digit pada baris ketiga.
2. Modulokan hasil sebelumnya dengan mod 10.
3. Kurangkan bilangan 10 dengan hasil modulo tersebut sehingga diperoleh *check digit*.

Algoritma Luhn ini juga memuat langkah-langkah dalam mengecek validitas dari suatu nomor identifikasi. Proses validasi suatu nomor akun yang terdefinisi bisa dilakukan sebagai berikut.

1. Kalikan setiap digit genap dengan 2 dari ujung kanan.
2. Untuk setiap hasil perkalian yang bernilai > 9 , maka kurangkan hasil perkalian tersebut dengan 9.
3. Jumlahkan semua digit-digit satu per satu seperti langkah saat pembentukan *check digit* sebelumnya.
4. Jika penjumlahan semua digit habis dibagi 10, maka nomor akun tersebut valid.

Berikat ini adalah implementasi dari pengecekan validitas dari suatu nomor akun dengan algoritma checkLuhn.

```
function checkLuhn(string purportedCC) {
    int sum :=
    integer(purportedCC[length(purportedCC)-1])
    int nDigits := length(purportedCC)
    int parity := nDigits modulus 2
    for i from 0 to nDigits - 2 {
        int digit :=
        integer(purportedCC[i])
        if i modulus 2 = parity
            digit := digit * 2
        if digit > 9
            digit := digit - 9
        sum := sum + digit
    }
    return (sum modulus 10) = 0
}
```

D. Nomor IMEI

1. Definisi IMEI dan Kegunaannya

IMEI (International Mobile Equipment Identity) adalah angka unik yang tertera pada setiap handphone sebagai identitas dari suatu handphone. Setiap handphone memiliki IMEI masing-masing mengingat IMEI sendiri adalah nomor yang menunjukkan identitas suatu handphone.

IMEI ini bersifat seperti kode produksi, dengan ketentuan setiap handphone mempunyai kode produksi masing-masing, yaitu IMEI. Dengan adanya IMEI, pengguna dapat mengetahui keaslian handphone dengan mengecek kevalidan dari nomor IMEI tersebut.



Gambar 2. IMEI dalam kardus handphone (Sumber : <https://imei.info/faq>)

Nomor IMEI dapat ditemukan dari box kardus handphone tersebut, label pada handphone, atau pengecekan secara digital menggunakan dial number. Berikut ini adalah beberapa hal yang bisa diidentifikasi dari suatu nomor IMEI.

1. Jaringan GSM dan negara handphone berasal.
2. Informasi terkait jaminan dan garansi handphone.
3. Informasi terkait versi sistem dan spesifikasi.



Gambar 3. Contoh IMEI Handphone (Sumber : Penulis)

Nomor IMEI digunakan oleh jaringan GSM untuk mengidentifikasi perangkat yang valid sehingga ketika handphone tersebut dicuri atau hilang, provider GSM dapat memutuskan jaringan terhadap handphone tersebut agar tidak bisa mengakses jaringan di negara tersebut. Nomor telepon seluler MEI yang terhubung ke jaringan GSM disimpan dalam basis data EIR (*Equipment Identity Register*) yang berisi semua peralatan ponsel yang valid.

Nomor IMEI juga bisa dicek dengan pemanggilan menggunakan *dial number* memasukkan *#06# (star-pound-zero-six-pound) menggunakan keypad. Selain itu, nomor IMEI juga bisa ditemukan dalam kompartemen baterai. Ketika terjadi kasus pencurian handphone atau handphone hilang, pemilik dapat menghubungi CEIR (*Central Equipment Identity Register*) dan melaporkan untuk pemblokiran handphone tersebut. Setelah itu, ponsel tidak dapat digunakan kembali meskipun kartu SIM ponsel sudah diganti.



Gambar 4. Contoh pemanggilan IMEI secara pemanggilan (Sumber : Penulis)

2. Struktur IMEI

Nomor IMEI dibentuk berdasarkan algoritma Luhn. Umumnya, nomor IMEI berbentuk 17 digit atau 15 digit angka. IMEI memiliki format atau struktur sebagai berikut.

Format 15 digit : AA – BBBB – CCCCC – D

1. AA: *Reporting Body Identifier*
2. BBBB : *The remainder of TAC (FAC)*
3. CCCCC : *Serial sequence of model (SNR)*
4. D : *Luhn check digit* untuk melakukan validasi terhadap nomor tersebut, yaitu AA – BBBB – CCCCC.

III. HASIL DAN PEMBAHASAN

Seperti yang telah dijelaskan sebelumnya, IMEI merupakan *sequence* angka yang merupakan kombinasi 15-digit dari bilangan bulat 0–9. Sebuah nomor IMEI dikatakan valid ketika dalam pengujian algoritma Luhn menghasilkan hasil yang valid, yaitu check digit sesuai dan membuat total digit habis dibagi 10.

Mengacu pada algoritma Luhn yang telah dijelaskan sebelumnya, berikut ini adalah implementasi algoritma tersebut dalam bentuk source code program dengan bahasa C.

```
#include <stdio.h>
#include "string.h"
#include "boolean.h"

/* Fungsi Luhn's Algorithm */
boolean checkLuhn (char IMEINumber[])
{
    /* Kamus Lokal */
    int sum = 0;
    int nDigits = strlen(IMEINumber);
    int digitvalue, i;
    /* Algoritma */
    for (i = 0; i <= nDigits-1; i++){
        if (i % 2 == 1){
            digitvalue = IMEINumber[i] - '0';
            digitvalue = digitvalue * 2;
            if (digitvalue > 9){
                digitvalue = digitvalue - 9;
            }
        }
        else {
            digitvalue = IMEINumber[i] - '0';
        }
        sum = sum + digitvalue;
    }
    return (sum % 10 == 0);
}

/* Main Program */
int main()
{
    /* Kamus Lokal */
    char IMEINumber[100];
    /* Algoritma */
    printf("Masukkan nomor IMEI yang ingin diperiksa : ");
    scanf("%s", &IMEINumber);
    if (checkLuhn(IMEINumber)){
        printf("IMEI %s valid\n", IMEINumber);
    }
    else {
        printf("IMEI %s not valid\n", IMEINumber);
    }
    return 0;
}
```

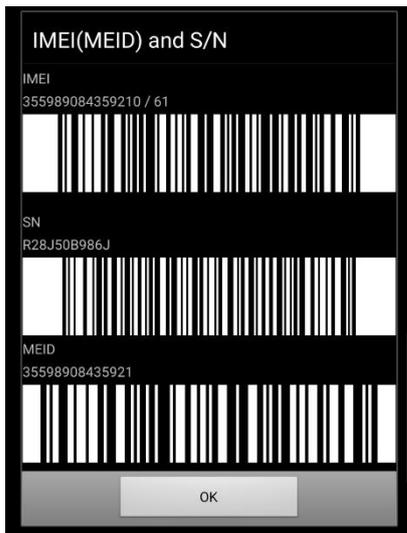
Gambar 5. Source Code Program Algoritma Luhn (Sumber : Penulis)

Pada bagian hasil dan pembahasan ini, penulis akan menguji dua buah kasus uji sebagai berikut.

1. IMEI : 355989084359210
IMEI pertama adalah IMEI yang diperoleh dari pemanggilan secara *dial number* pada Samsung Galaxy S8+
2. IMEI : 493736800637052
IMEI kedua adalah IMEI yang diperoleh dari <https://dyrk.org/tools/imei/> dengan mengubah digit validasi dengan digit validasi lain.

1. Percobaan Pertama

Berikut ini adalah IMEI dari suatu handphone yang akan kami cek kevalidannya.



Gambar 6. IMEI Handphone (Sumber : Penulis)

Pada gambar diatas, nomor IMEI dari handphone tersebut adalah 355989084359210. Secara struktur, nomor IMEI tersebut dapat dijelaskan dengan bagian-bagian sebagai berikut.

35 – 598908 – 435921 – 0

1. AA – 35 - Reporting Body Identifier
2. BBBB – Remainder TAC - 598908
3. CCCCC – SNR - 435921
4. D – 0 – digit Luhn

Nomor IMEI akan dituliskan dalam bentuk baris seperti berikut.

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

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

Bagian yang diberikan warna kuning akan dikalikan dua sehingga menghasilkan digit-digit yang baru pada segmen elemen tersebut.

3	5	5	9	8	9	0	8	4	3	5	9	2	1	0
3	10	5	18	8	18	0	16	4	6	5	18	2	2	0
3	1	5	9	8	9	0	7	4	6	5	9	2	2	0

Untuk hasil perkalian digit posisi genap yang lebih dari 9, lakukan pengurangan dengan 9 sehingga diperoleh baris terakhir. Kemudian, digit-digit baru pada baris terakhir dijumlahkan sehingga diperoleh $3 + 1 + 0 + 5 + 1 + 8 + 8 + 1 + 8 + 0 + 1 + 6 + 4 + 6 + 5 + 1 + 8 + 2 + 2 = 70$.

Hasil penjumlahan bernilai 70. Kemudian hasil penjumlahan tersebut dioperasikan dengan modulo 10 sehingga diperoleh

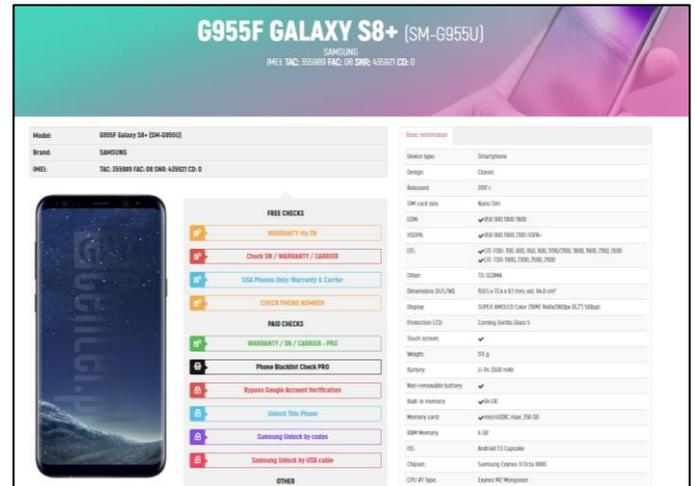
$$70 \text{ mod } 10 = 0$$

Karena hasil operasi tersebut bernilai 0 atau 70 habis dibagi 10, maka nomor IMEI handphone tersebut valid sehingga handphone dengan IMEI 355989084359210 adalah asli.

Jika menggunakan aplikasi pengecekan langsung dari <https://imei.info>, maka akan diperoleh gambar sebagai berikut.



Gambar 7. IMEI Calculator (Sumber: <https://imei.info>)



Gambar 8. Hasil pengecekan IMEI (Sumber: <https://imei.info>)

Dari gambar tersebut, spesifikasi handphone, merk handphone, dan IMEI sesuai dengan informasi yang terdaftar pada <https://imei.info> sehingga handphone dengan IMEI 355989084359210 adalah asli, yaitu Samsung Galaxy S8+.

Jika dicek menggunakan program yang telah dibuat, maka hasil yang dikeluarkan oleh program adalah sebagai berikut.

Masukkan nomor IMEI yang ingin diperiksa : 355989084359210
IMEI 355989084359210 valid

Gambar 9. Hasil eksekusi program (Sumber: Penulis)

2. Percobaan Kedua

Berikut ini adalah percobaan yang ke-2, yaitu sebuah nomor IMEI yang diduga tidak valid atau palsu. Salah satu contoh IMEI yang tidak valid adalah 493736800637052. Secara struktur, nomor IMEI tersebut dapat dijelaskan dengan bagian-bagian sebagai berikut.

49 – 373680 – 063705 – 2

1. AA – 49 - Reporting Body Identifier
2. BBBB – Remainder TAC - 373680
3. CCCCC – SNR - 063705
4. D – 2 – digit Luhn

Nomor IMEI akan dituliskan dalam bentuk baris seperti berikut. 493736800637052

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

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

Bagian yang diberikan warna kuning akan dikalikan dua sehingga menghasilkan digit-digit yang baru pada segmen elemen tersebut.

4	9	3	7	3	6	8	0	0	6	3	7	0	5	2
4	18	3	14	3	12	8	0	0	12	3	14	0	10	2
4	9	3	5	3	3	8	0	0	3	3	5	0	1	2

Untuk hasil perkalian digit posisi genap yang lebih dari 9, lakukan pengurangan dengan 9 sehingga diperoleh baris terakhir. Kemudian, digit-digit baru pada baris terakhir dijumlahkan sehingga diperoleh $4 + 9 + 3 + 5 + 3 + 3 + 8 + 0 + 0 + 3 + 3 + 5 + 0 + 1 + 2 = 49$.

Hasil penjumlahan bernilai 49. Hasil penjumlahan ini akan dioperasikan dengan modulo 10 sehingga diperoleh $49 \bmod 10 = 9$

Perhatikan bahwa hasil penjumlahan tidak habis dibagi 10 atau bersisa 9 sehingga nomor IMEI tersebut terbukti tidak valid dan tidak terdaftar pada <https://imei.info>. Berikut ini adalah gambar yang menunjukkan bahwa nomor IMEI tersebut adalah nomor yang tidak valid.



Gambar 10. IMEI yang tidak valid (Sumber: <https://imei.info>)

Jika dicek menggunakan program yang telah dibuat, maka hasil yang dikeluarkan oleh program adalah sebagai berikut.

```
Masukkan nomor IMEI yang ingin diperiksa : 493736800637052
IMEI 493736800637052 tidak valid
```

Gambar 11. Hasil eksekusi program (Sumber: Penulis)

Dari hasil percobaan diatas, dapat dinyatakan bahwa nomor IMEI memanfaatkan algoritma Luhn. Algoritma Luhn sendiri menerapkan prinsip modulo dalam mengecek validitas dari suatu nomor IMEI. Operasi modulo hanya dapat diterapkan bila bilangan a dan b adalah bilangan bulat sehingga nomor-nomor penyusun IMEI merupakan *sequence* bilangan bulat.

IV. KESIMPULAN

Teori bilangan terkait aritmatika modulo sering diaplikasikan dalam hal terkait pembuatan dan pengecekan nomor identifikasi atau *ID number*, seperti nomor identifikasi pada ponsel, yaitu IMEI. Setiap ponsel memiliki nomor IMEI yang unik satu sama lain. IMEI digunakan untuk mengecek keaslian dari ponsel. Prinsip validitas dari IMEI ini mengacu kepada operasi aritmatika modulo dengan implementasinya berupa algoritma Luhn.

V. UCAPAN TERIMA KASIH

Penulis mengucapkan puji dan syukur yang sebesar-besarnya kepada rahmat Tuhan Yang Maha Esa atas berkat dan rahmat-Nya sehingga makalah berjudul “Aplikasi Modulo dalam Pengecekan Keaslian IMEI” dapat diselesaikan. Penulis mengucapkan terima kasih kepada selaku dosen pengajar IF2120 Matematika Diskrit, Dra. Harlili M.Sc yang telah memberikan bimbingan dan ilmu terkait materi matematika diskrit ini, khususnya teori bilangan. Penulis juga mengucapkan terima kasih kepada teman-teman yang telah memberikan dukungan kepada penulis untuk menuliskan makalah ini. Tak lupa penulis juga mengucapkan terima kasih kepada para penulis sumber referensi yang telah memberikan penulis ilmu yang dibutuhkan untuk menyelesaikan makalah ini.

REFERENSI

- [1] Tim IMEI. <https://www.imei.info/>, diakses tanggal 1 Desember 2019, pukul 10.20 GMT+7.
- [2] ISO/IEC 7812-1:2017 Identification cards -- Identification of issuers -- Part 1: Numbering system
- [3] Munir, Rinaldi. Matematika Diskrit. Bandung: Informatika. 2016, edisi keenam.
- [4] Tim Penulis. <http://www.postmedya.com/teknologi/pengertian-fungsi-serta-struktur-nomor-imei-pada-ponsel/>. Diakses tanggal 1 Desember 2019, pukul 13.05 GMT+7.
- [5] Tim Penulis. <https://www.geeksforgeeks.org/luhn-algorithm/>. Diakses tanggal 30 November 2019, pukul 22.05 GMT+7
- [6] US Patent 2,950,048 – *Computer for Verifying Numbers*, Hans P Luhn, August 23, 1960

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, 6 Desember 2019

Michael Hans
13518056