

BILANGAN ACAK (RANDOM NUMBER)

Disajikan oleh:

Bernardus Budi Hartono

- Web : <http://pakhartono.wordpress.com/>
E-mail: pakhartono at gmail dot com
budihartono at acm dot org

Teknik Informatika [Gasal 2009 – 2010]

FTI - Universitas Stikubank Semarang

BILANGAN ACAK (RANDOM NUMBER)

Membangkitkan Bilangan Acak (Random)
Pseudo Random Number Generator

Bilangan acak yang berkualitas baik:

- a. bila terjadi perulangan atau munculnya bilangan acak yang sama → setelah sekian periode tertentu (semakin lama semakin baik)
- b. bila terjadi perulangan → kemunculannya tidak bisa diprediksi

Sumber bilangan acak

Sumber bilangan acak

- a. tabel bilangan random
- b. electronic random number
- c. pembangkitan bilangan acak semu (congruential pseudo random number generator) dengan algoritma matematika

Beberapa metode pembangkitan bilangan acak semu:

Additive (arithmetic) RNG

Multiplicative RNG (MRNG)

Mixed Congruential RNG

Metode Multiplicative RNG (MRNG) - 1

bilangan acak yang dibangkitkan oleh komputer (bersifat acak semu), dibangkitkan menggunakan rumus matematika yang dikerjakan berulang-ulang sesuai kebutuhan.

Metode Multiplicative RNG (MRNG)

Rumus

$$Z_{i+1} = (a \cdot Z_i + c) \text{ mod } m$$

$$R_1 = Z_{i+1} / m$$

bilangan random yang dihasilkan = $R_1, R_2, R_3, R_4, \dots$

Metode Multiplicative RNG (MRNG) - 2

$$Z_{i+1} = (a \cdot Z_i + c) \bmod m$$

$$R_1 = Z_{i+1} / m$$

ketentuan:

Memilih konstanta pengali (multiplier) a

Memilih Z_0

Z_0 bilangan ganjil yang besar

Untuk bilangan random pertama maka $Z_i \leftarrow Z_0$

Memilih c

c bilangan ganjil dan bukan kelipatan dari m

Memilih nilai m

catatan: pada proses iterasi, a, c, dan m, bersifat konstan

Metode Multiplicative RNG (MRNG) - 3

Contoh

carilah 3 bilangan acak menggunakan metode Multiplicative RNG, dengan nilai awal $Z_0=12357$, $a=19$, $c=237$, $m=128$.

Gunakan tingkat ketelitian 4 digit di belakang koma.

bilangan acak 1

$$\begin{aligned}Z_{i+1} &= (a \cdot Z_i + c) \bmod m \\ &= (19 \cdot 12357 + 237) \bmod 128 \\ &= 235020 \bmod 128 \\ &= 12\end{aligned}$$

$$\begin{aligned}R_1 &= Z_{i+1} / m \\ &= 12 / 128 \\ &= \mathbf{0.0938}\end{aligned}$$

Metode Multiplicative RNG (MRNG) - 4

bilangan acak 2

$$\begin{aligned}Z_{i+1} &= (a \cdot Z_i + c) \bmod m \\ &= (19 \cdot 12 + 237) \bmod 128 \\ &= 465 \bmod 128 \\ &= 81\end{aligned}$$

$$\begin{aligned}R_1 &= Z_{i+1} / m \\ &= 81 / 128 \\ &= \mathbf{0.6328}\end{aligned}$$

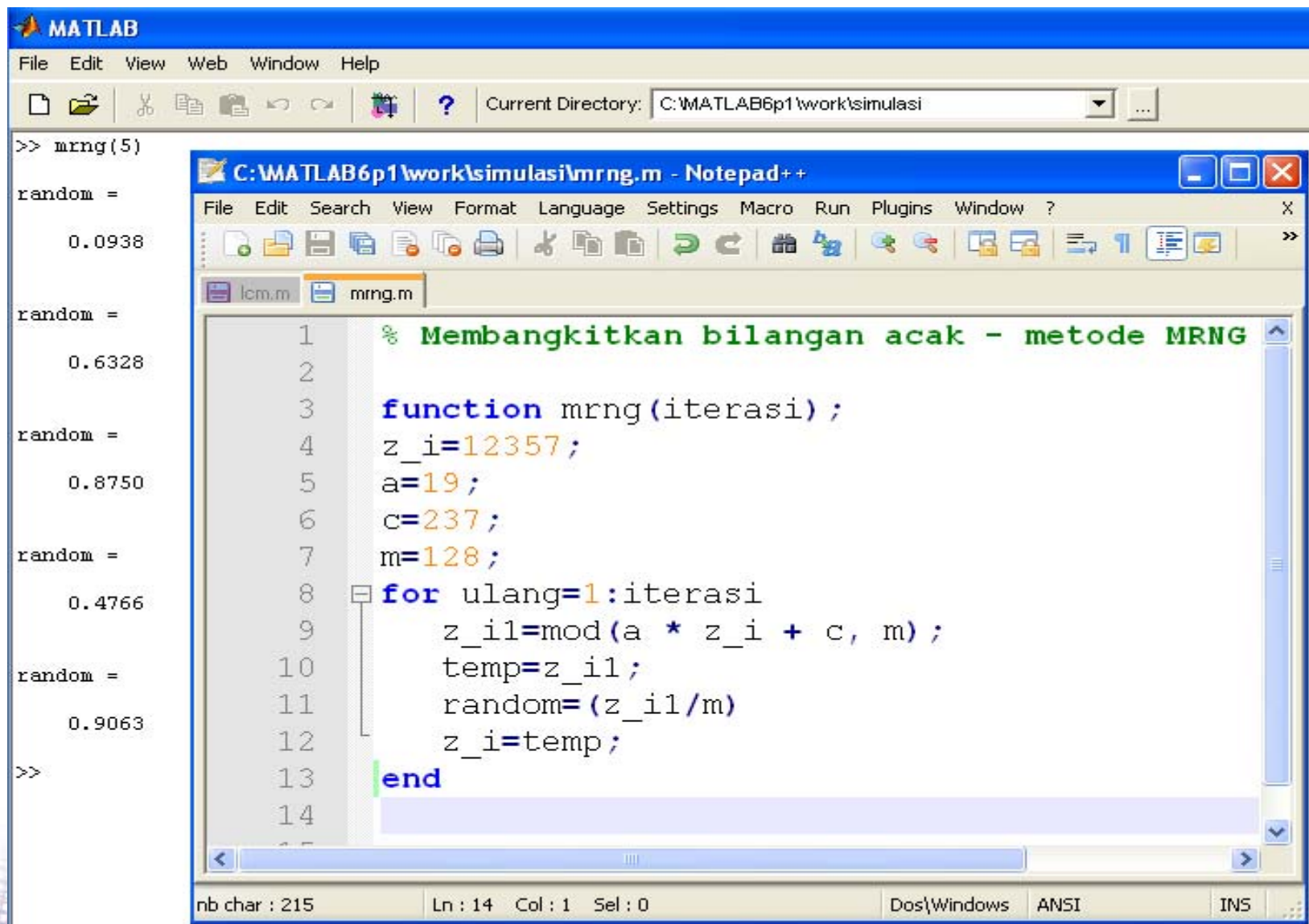
bilangan acak 3

$$\begin{aligned}Z_{i+1} &= (a \cdot Z_i + c) \bmod m \\ &= (19 \cdot 81 + 237) \bmod 128 \\ &= 1776 \bmod 128 \\ &= 112\end{aligned}$$

$$\begin{aligned}R_1 &= Z_{i+1} / m \\ &= 112 / 128 \\ &= \mathbf{0.875}\end{aligned}$$

Metode Multiplicative RNG (MRNG) - 5

Contoh program (Matlab) dan hasilnya



The image shows a MATLAB window on the left and a Notepad++ window on the right. The MATLAB window displays the execution of the `mrng(5)` function, resulting in five random numbers: 0.0938, 0.6328, 0.8750, 0.4766, and 0.9063. The Notepad++ window shows the source code for the `mrng` function, which implements the Multiplicative Random Number Generator (MRNG) method. The code includes a comment in Indonesian, a function definition, and a loop that iterates 5 times, generating random numbers using the formula $z_{i+1} = \text{mod}(a * z_i + c, m)$.

```
>> mrng(5)

random =

    0.0938

random =

    0.6328

random =

    0.8750

random =

    0.4766

random =

    0.9063

>>
```

```
C:\MATLAB6p1\work\simulasi\mrng.m - Notepad++
File Edit Search View Format Language Settings Macro Run Plugins Window ?
lcm.m mrng.m
1  % Membangkitkan bilangan acak - metode MRNG
2
3  function mrng(iterasi);
4  z_i=12357;
5  a=19;
6  c=237;
7  m=128;
8  for ulang=1:iterasi
9      z_i1=mod(a * z_i + c, m);
10     temp=z_i1;
11     random=(z_i1/m)
12     z_i=temp;
13 end
14
nb char : 215 Ln : 14 Col : 1 Sel : 0 Dos\Windows ANSI INS
```


Metode Linear Congruent Method (LCM) - 1

Rumus

$$Z_{i+1} = (a \cdot Z_i + c) \bmod m$$

bilangan random yang dihasilkan = $Z_1, Z_2, Z_3, Z_4, \dots$

Metode ini banyak digunakan di dalam program komputer, ketentuan untuk memilih $Z_0, a, c,$ dan m sama dengan metode MRNG.

Pada proses iterasi, $a, c,$ dan $m,$ bersifat konstan.

Disini meniadakan langkah $R_1 = Z_{i+1} / m,$ sehingga bilangan acak yang dihasilkan adalah bulat dan bernilai $< m.$

Metode Linear Congruent Method (LCM) - 2

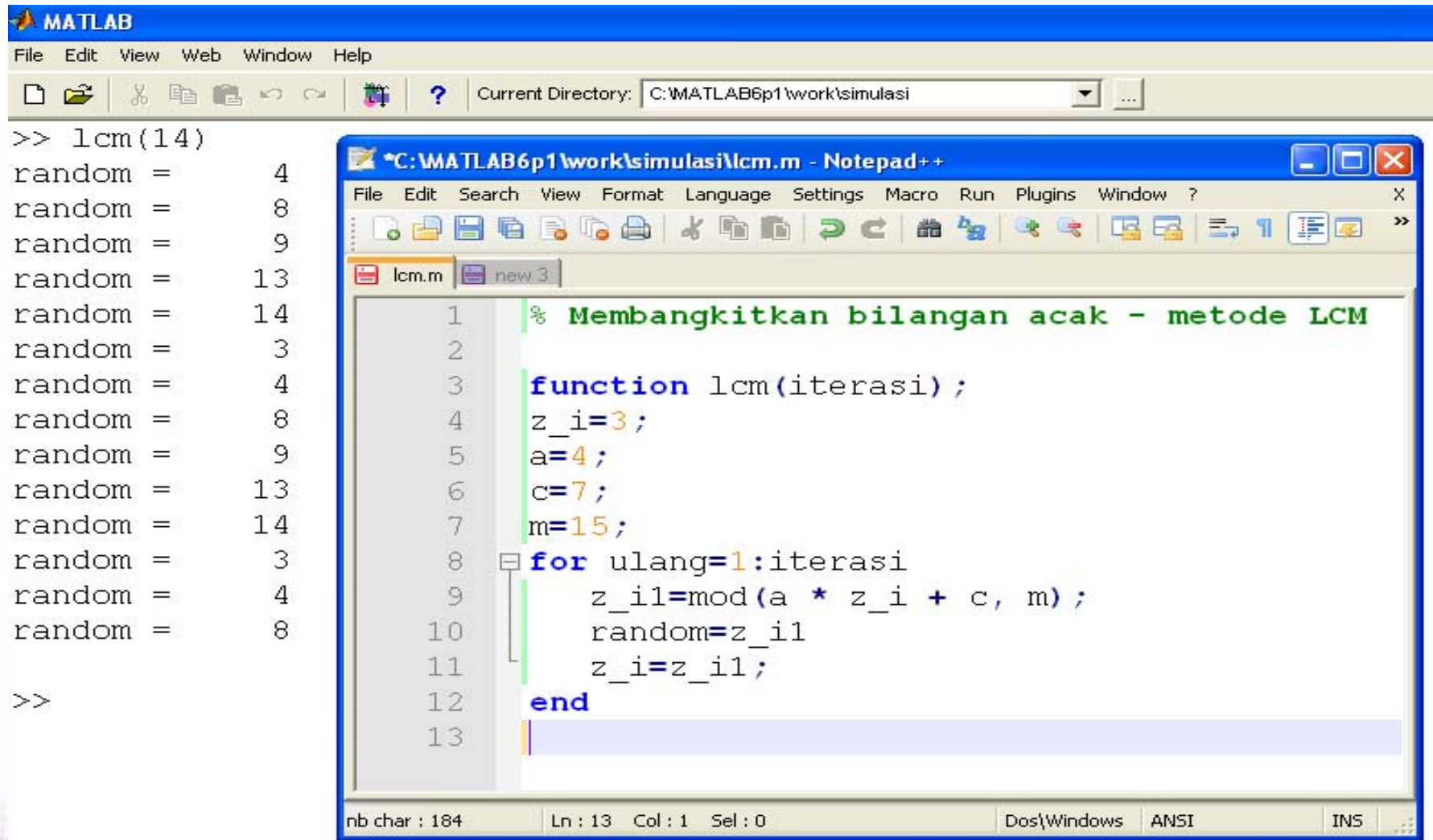
Contoh1 LCM

Bangkitkanlah 14 buah bilangan acak menggunakan metode LCM, dengan nilai awal $Z_0=3$, $a=4$, $c=7$, $m=15$.

Contoh program (Matlab) dan Hasil bilangan acak dapat dilihat pada gambar berikut ini.

Metode Linear Congruent Method (LCM) - 3

Contoh program (Matlab) dan hasilnya



The image shows a MATLAB window on the left and a Notepad++ window on the right. The MATLAB window displays the execution of the `lcm(14)` function, which generates a sequence of random numbers. The Notepad++ window shows the source code for the `lcm.m` file, which implements the Linear Congruent Method (LCM) algorithm. The code includes a function definition, variable initialization, and a loop that iterates 14 times, updating the random number `z_i` based on the formula $z_{i+1} = \text{mod}(a * z_i + c, m)$.

```
>> lcm(14)
random = 4
random = 8
random = 9
random = 13
random = 14
random = 3
random = 4
random = 8
random = 9
random = 13
random = 14
random = 3
random = 4
random = 8
>>
```

```
*C:\MATLAB6p1\work\simulasi\lcm.m - Notepad++
File Edit Search View Format Language Settings Macro Run Plugins Window ?
lcm.m new 3
1 % Membangkitkan bilangan acak - metode LCM
2
3 function lcm(iterasi);
4 z_i=3;
5 a=4;
6 c=7;
7 m=15;
8 for ulang=1:iterasi
9     z_i1=mod(a * z_i + c, m);
10    random=z_i1
11    z_i=z_i1;
12 end
13
```

nb char : 184 Ln : 13 Col : 1 Sel : 0 Dos\Windows ANSI INS

Metode Linear Congruent Method (LCM) - 4

Contoh1 LCM

Bangkitkanlah 14 buah bilangan acak menggunakan metode LCM, dengan nilai awal $Z_0=3$, $a=4$, $c=7$, $m=15$.

Bilangan acak yang dihasilkan:

4 8 9 13 14 3 4 8 9 13 14 3 4 8

Analisa:

Kemunculan bilangan acak akan berulang setelah membangkitkan enam buah bilangan acak

Metode Linear Congruent Method (LCM) - 5

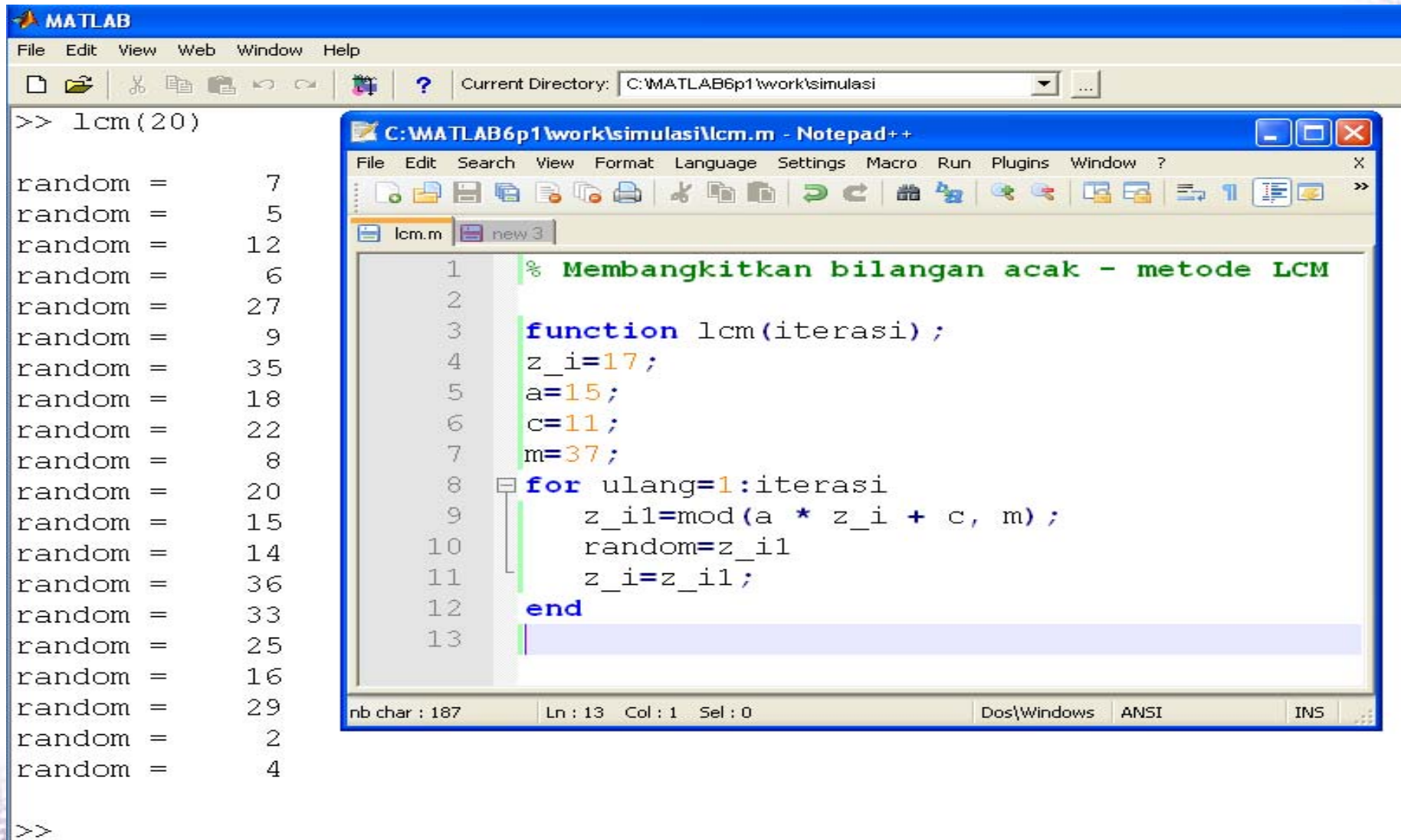
Contoh2 LCM

Bangkitkanlah 20 buah bilangan acak menggunakan metode LCM, dengan nilai awal $Z_0=17$, $a=15$, $c=11$, $m=37$.

Contoh program (Matlab) dan Hasil bilangan acak dapat dilihat pada gambar berikut ini.

Metode Linear Congruent Method (LCM) - 6

Contoh program (Matlab) dan hasilnya



The image shows a MATLAB window on the left and a Notepad++ window on the right. The MATLAB window displays the execution of the `lcm(20)` function, which generates a sequence of random numbers. The Notepad++ window shows the source code for the `lcm` function, which implements the Linear Congruent Method (LCM) with parameters `a=15`, `c=11`, and `m=37`.

```
>> lcm(20)

random =      7
random =      5
random =     12
random =      6
random =     27
random =      9
random =     35
random =     18
random =     22
random =      8
random =     20
random =     15
random =     14
random =     36
random =     33
random =     25
random =     16
random =     29
random =      2
random =      4

>>
```

```
C:\MATLAB6p1\work\simulasi\lcm.m - Notepad++
File Edit Search View Format Language Settings Macro Run Plugins Window ?
lcm.m new 3
1  % Membangkitkan bilangan acak - metode LCM
2
3  function lcm(iterasi);
4  z_i=17;
5  a=15;
6  c=11;
7  m=37;
8  for ulang=1:iterasi
9      z_i1=mod(a * z_i + c, m);
10     random=z_i1
11     z_i=z_i1;
12 end
13
```

nb char : 187 Ln : 13 Col : 1 Sel : 0 Dos\Windows ANSI INS

Metode Linear Congruent Method (LCM) - 7

Contoh1 LCM

Bangkitkanlah 20 buah bilangan acak menggunakan metode LCM, dengan nilai awal $Z_0=17$, $a=15$, $c=11$, $m=37$.

Bilangan acak yang dihasilkan:

7, 5, 12, 6, 27, 9, 35, 18, 22, 8, 20, 15, 14, 36, 33, 25, 16, 29, 2, 4

Analisa:

Kemunculan bilangan acak tidak (belum) berulang untuk pembangkitan 20 buah bilangan acak yang pertama.

Bilangan acak yang mungkin muncul adalah [0....36].

Kesimpulan.....

*Untuk kedua algoritma MRNG dan LCM.....
penentuan nilai awal Z_0 dan konstanta (a , c , dan m) akan
menentukan kualitas bilangan acak yang dihasilkan.*

*Bilangan acak yang baik (pada umumnya).....
apabila terjadinya perulangan atau munculnya bilangan acak
yang sama, dapat terjadi setelah sekian banyak
pembangkitan bilangan acak (semakin banyak akan semakin
baik) serta tidak bisa diprediksi kapan terjadi
perulangannya.*

Referensi.....

- *“Pengantar Sistem Simulasi”*, Thomas J. Kakiay, Penerbit Andi, Yogyakarta, 2004.
- *“Simulasi Teori dan Aplikasinya”*, Bonett Satya Lelono Djati, Penerbit Andi, Yogyakarta, 2007.