



# **Membangkitkan Bilangan Acak Menggunakan Matlab**

**Achmad Basuki**

2004



# Materi

- Linear Congruent Method
- Metode Resuffle
- Fungsi Standard Membangkitkan Bilangan Acak
- Grafik dan Statistik Bilangan Acak



# Pseudo Random Number

- Bilangan acak yang dibangkitkan oleh komputer merupakan bilangan acak semu, karena pembangkitannya menggunakan operasi-operasi aritmatika.
- Banyak algoritma atau metode yang dapat digunakan untuk membangkitkan bilangan acak.



# Linear Congruent Method

- Linear Congruent Method (LCM) merupakan metode pembangkitkan bilangan acak yang banyak digunakan dalam program komputer.
- LCM memanfaatkan model linier untuk membangkitkan bilangan acak yang didefinisikan dengan:

$$x_{n+1} = ( a x_n + c ) \text{ mod } m$$

Dimana :  $x_n$  = adalah bil. acak ke n  
a dan c adalah konstanta LCM  
m adalah batas maksimum bilangan acak



# Contoh 1

Membangkitkan bilangan acak sebanyak 8 kali dengan  $a=2$ ,  $b=7$ ,  $m = 10$  dan  $x(0)=2$

$$x(1) = ( 2 (2) + 7 ) \text{ mod } 10 = 1$$

$$x(2) = ( 2 (1) + 7 ) \text{ mod } 10 = 9$$

$$x(3) = ( 2 (9) + 7 ) \text{ mod } 10 = 5$$

$$x(4) = ( 2 (5) + 7 ) \text{ mod } 10 = 7$$

$$x(5) = ( 2 (7) + 7 ) \text{ mod } 10 = 1$$

$$x(6) = ( 2 (1) + 7 ) \text{ mod } 10 = 9$$

$$x(7) = ( 2 (9) + 7 ) \text{ mod } 10 = 5$$

$$x(8) = ( 2 (5) + 7 ) \text{ mod } 10 = 7$$

Bilangan acak yang dibangkitkan adalah :

1 9 5 7 1 9 5 7

 Terjadi pengulangan bilangan secara periodik (4)



# Contoh 2

Membangkitkan bilangan acak sebanyak 8 kali dengan  $a=4$ ,  $b=7$ ,  $m = 15$  dan  $x(0)=3$

$$x(1) = ( 4 (3) + 7 ) \text{ mod } 15 = 4$$

$$x(2) = ( 4 (4) + 7 ) \text{ mod } 15 = 8$$

$$x(3) = ( 4 (8) + 7 ) \text{ mod } 15 = 5$$

$$x(4) = ( 4 (5) + 7 ) \text{ mod } 15 = 12$$

$$x(5) = ( 4 (12) + 7 ) \text{ mod } 15 = 10$$

$$x(6) = ( 4 (10) + 7 ) \text{ mod } 15 = 2$$

$$x(7) = ( 4 (2) + 7 ) \text{ mod } 15 = 0$$

$$x(8) = ( 4 (0) + 7 ) \text{ mod } 15 = 7$$

Bilangan acak yang dibangkitkan adalah :

4 8 5 12 10 2 0 7



Tidak terlihat pengulangan bilangan secara periodik



# Contoh 3

Membangkitkan bilangan acak sebanyak 16 kali dengan  $a=4$ ,  $b=7$ ,  $m = 15$  dan  $x(0)=3$

$x(1) = (4(3) + 7) \bmod 15 = 4$	$x(9) = (4(7) + 7) \bmod 15 = 13$
$x(2) = (4(4) + 7) \bmod 15 = 8$	$x(10) = (4(13) + 7) \bmod 15 = 14$
$x(3) = (4(8) + 7) \bmod 15 = 5$	$x(11) = (4(14) + 7) \bmod 15 = 3$
$x(4) = (4(5) + 7) \bmod 15 = 12$	$x(12) = (4(3) + 7) \bmod 15 = 4$
$x(5) = (4(12) + 7) \bmod 15 = 10$	$x(13) = (4(4) + 7) \bmod 15 = 8$
$x(6) = (4(10) + 7) \bmod 15 = 2$	$x(14) = (4(8) + 7) \bmod 15 = 5$
$x(7) = (4(2) + 7) \bmod 15 = 0$	$x(15) = (4(5) + 7) \bmod 15 = 12$
$x(8) = (4(0) + 7) \bmod 15 = 7$	$x(16) = (4(12) + 7) \bmod 15 = 10$

Bilangan acak yang dibangkitkan adalah :

4 8 5 12 10 2 0 7 13 14 3 4 8 5 12 10



Terlihat pengulangan bilangan secara periodik (10)



# Kesimpulan Pada LCM

- Terjadi pengulangan pada periode waktu tertentu atau setelah sekian kali pembangkitan, hal ini adalah salah satu sifat dari metode ini, dan *pseudo random generator* pada umumnya.
- Penentuan konstanta LCM ( $a$ ,  $c$  dan  $m$ ) sangat menentukan baik tidaknya bilangan acak yang diperoleh dalam arti memperoleh bilangan acak yang seakan-akan tidak terjadi pengulangan.



# Membuat Fungsi Pembangkit Bilangan Acak Dengan LCM

Tuliskan program fungsi di bawah ini dan simpan dalam nama file LCM.m

```
function x=LCM(xs)
% Membangkitkan bilangan acak dengan LCM
a=23; c=15; m=257;
x=mod(a*xs+c,m);
```

*Fungsi ini menghasilkan satu bilangan  $x$ , dengan memasukkan  $x$  sebelumnya ( $xs$ ), sedangkan  $a$ ,  $c$  dan  $m$  merupakan konstanta yang harus didefinisikan*



# Memanggil Bilangan Acak Dengan Fungsi LCM

Membangkitkan 4 bilangan acak dengan  $x(0) = 10$  adalah sebagai berikut:

```
>> LCM(10)
ans =
    245
>> LCM(245)
ans =
    253
>> LCM(253)
ans =
    180
>> LCM(180)
ans =
    43
```



# Memanggil Bilangan Acak Dengan Fungsi LCM

Membangkitkan 20 bilangan acak dengan  $x(0) = 150$  adalah sebagai berikut:

```
>> xs=150;
>> for i=1:20
x(i)=LCM(xs);
xs=x(i);
end
>> x
x =
Columns 1 through 6
 124   40  164  189  250  111
Columns 7 through 12
 255  226   73  152  170   70
Columns 13 through 18
   83  125   63  179   20  218
Columns 19 through 20
 146   32
```



# Tugas 1

Carilah nilai  $a$ ,  $c$  dan  $m$  pada LCM agar dalam 100 kali pembangkitan bilangan acak belum terlihat adanya pengulangan



# Metode Resuffle

- Metode ini melakukan pergantian posisi pada bilangan acak dengan meng-update nilai bilangan acak pada posisi yang diganti.
- Metode Resuffle digunakan untuk memperbaiki bilangan acak dari LCM dengan menghilangkan sifat periodik yang ada pada bilangan acak.



# Algoritma dari Metode Resuffle

1. Bangkitkan  $n$  bilangan acak  $a_i [0,1]$  dengan LCM
2. Bangkitkan bilangan acak baru  $b [0,1]$  dengan LCM
3. Hitung  $k = n \times b$
4. Hitung  $r = a_k$
5. Hitung  $a_k = b$
6. Ambil nilai  $r$  sebagai sebagai bilangan acak baru dan ulangi langkah 2.



# Contoh Program Resuffle

```
% Nilai pembangkitan
% dan state awal
a=4; c=1; m=9; r(1)=3;

% membangkitkan 9 bil acak
% state awal dianggap 1 bil
for i=1:9
r(i+1)=mod(LCM(r(i)),10);
end

disp('Sebelum resuffle:')
disp(r)
```

```
% resuffle 3 kali
for i=1:3
r(10+i)=mod(a*r(9+i)+c,m);
p=r(10+i);
r(p+1)=p;
end

disp('Sesudah resuffle:')
```

Simpan program ini dalam file  
resuffle.m



# Contoh Hasil Resuffle

```
>> reshuffle
```

Sebelum reshuffle:

3 4 8 6 7 2 0 1 5 3

Sesudah reshuffle:

3 4 8 6 4 2 6 1 8 3

Bilangan yang di-reshuffle



# Fungsi Standard Untuk Membangkitkan Bilangan Acak

- Fungsi untuk membangkitkan bilangan acak dalam Matlab adalah **rand()**.
- Fungsi **rand(m,n)** membangkitkan bilangan acak 0 s/d 1, sebanyak m baris dan n kolom.
- Fungsi **rand('state',a)** digunakan untuk mengganti nilai state awal bilangan acak atau  $x(0)$  dalam LCM.



# Contoh Membangkitkan Bilangan Acak Dengan Matlab

- Membangkitkan 10 bilangan acak 0 s/d 1 dapat digunakan fungsi **rand(1,10)**
- Membangkitkan matrik dengan ukuran 4x6 yang elemennya bilangan acak 0 s/d 1 dapat digunakan fungsi **rand(4,6)**
- Membangkitkan 10 bilangan acak bulat 0 s/d 7 digunakan fungsi **floor(8\*rand(1,10))**
- Membangkitkan 10 bilangan acak antara 4 s/d 10 digunakan fungsi **floor(7\*rand(1,10))+4**
- Membangkitkan matrik 3x5 yang elemennya acak dari 0 s/d 7 digunakan fungsi **floor(8\*rand(3,5))**



# Tugas 2

Tuliskan fungsi atau perintah dalam Matlab untuk membangkitkan:

- (1) Bilangan acak ganjil antara 0 s/d 10
- (2) Bilangan acak antara 0 s/d 10 kecuali bilangan 6 dan 7



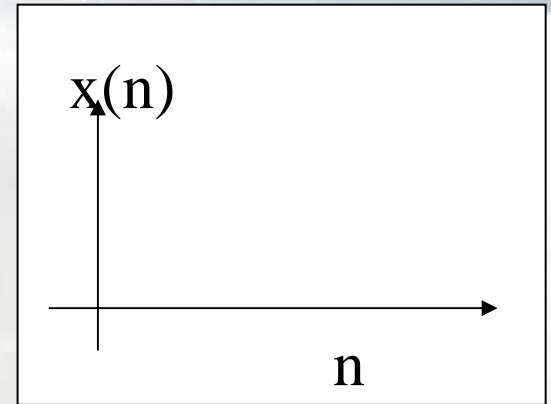
# Grafik dan Statistik Bilangan Acak

- Plot
- Histogram
- PDF
- CDF
- Analisa Data



# PLOT

Menampilkan semua bilangan acak



```
>> x=rand(1,8)
```

```
x =
```

```
Columns 1 through 3
```

```
0.4387 0.4983 0.2140
```

```
Columns 4 through 6
```

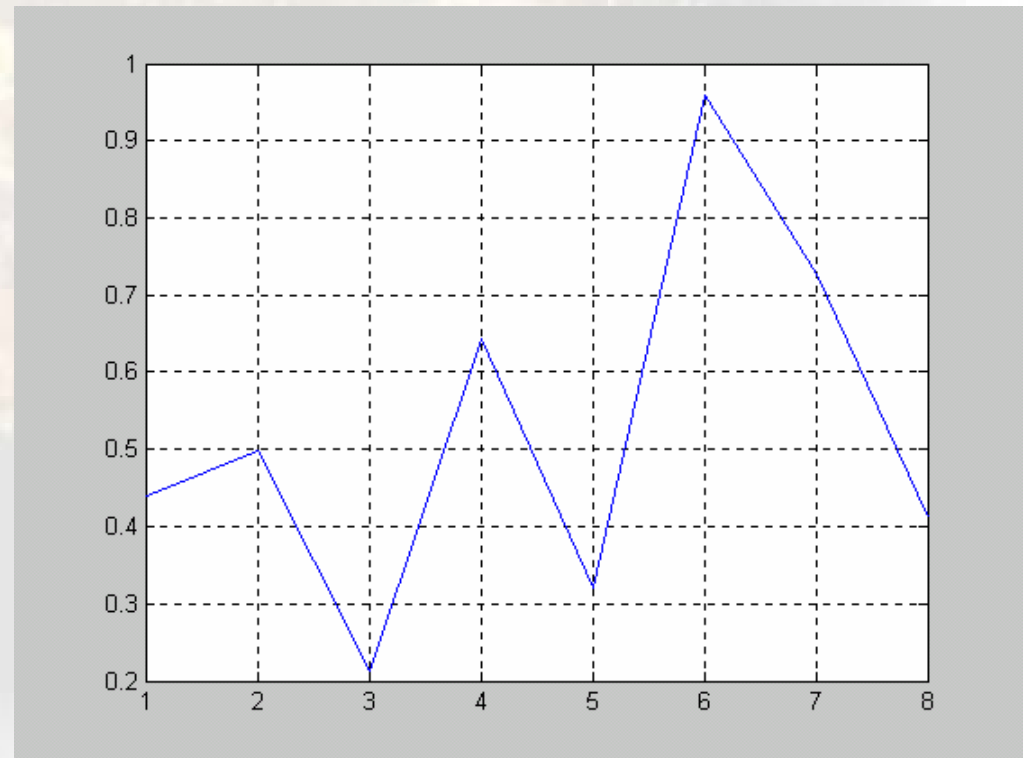
```
0.6435 0.3200 0.9601
```

```
Columns 7 through 8
```

```
0.7266 0.4120
```

```
>> plot(x),grid
```

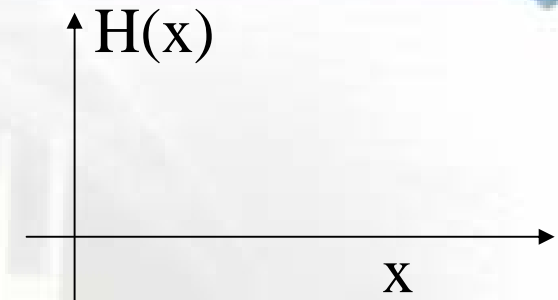
```
>>
```





# Histogram

Menampilkan histogram dari bilangan acak



```
>> x=floor(rand(1,15)*10)
```

```
x =
```

```
Columns 1 through 6
```

```
7 2 4 9 6 2
```

```
Columns 7 through 12
```

```
8 6 1 2 6 6
```

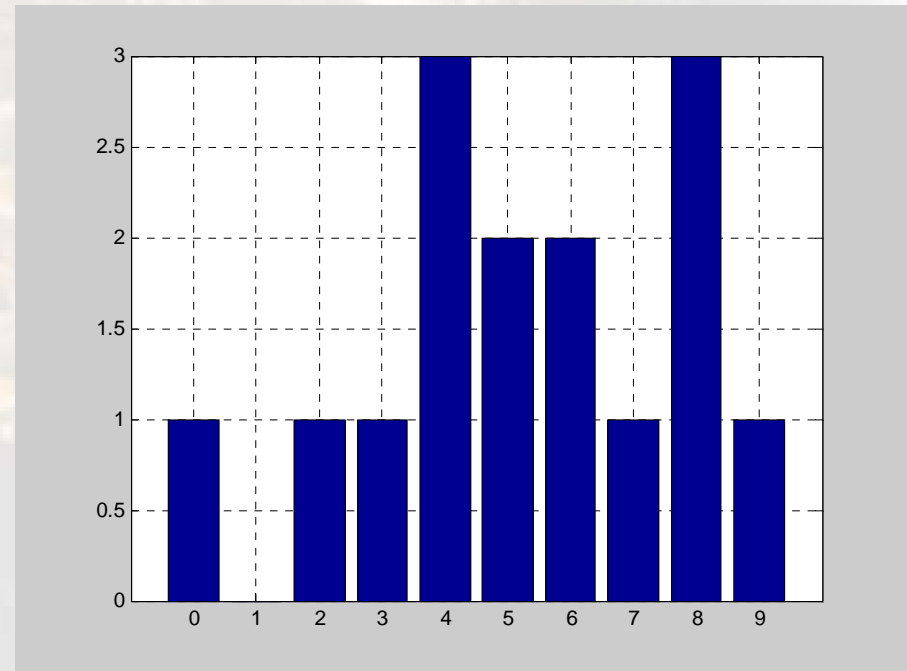
```
Columns 13 through 15
```

```
3 5 4
```

```
>> t=0:9;
```

```
>> h=hist(x,t);
```

```
>> bar(t,h), grid
```

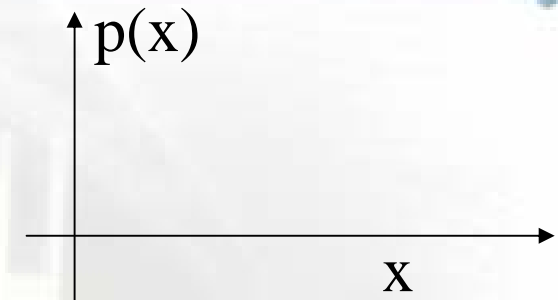




# PDF

Menampilkan fungsi kepadatan probabilitas (probability density function) dari bilangan acak

$$p(x_i) = \frac{H(x_i)}{\sum_{j=1}^n H(x_j)}$$



```
>> x=floor(10*rand(1,16))
```

```
x =
```

```
Columns 1 through 6
```

```
6 7 6 0 4 4
```

```
Columns 7 through 12
```

```
3 1 6 6 7 4
```

```
Columns 13 through 16
```

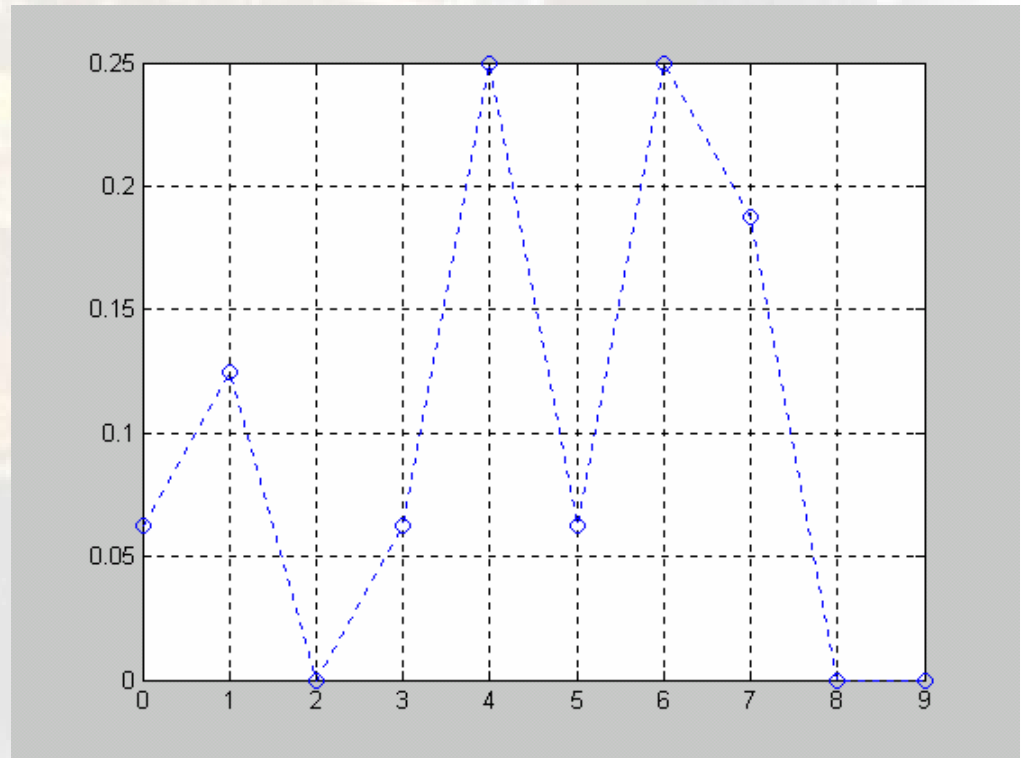
```
5 1 4 7
```

```
>> t=0:9;
```

```
>> h=hist(x,t);
```

```
>> p=h/sum(h);
```

```
>> plot(t,p,'o:'),grid
```

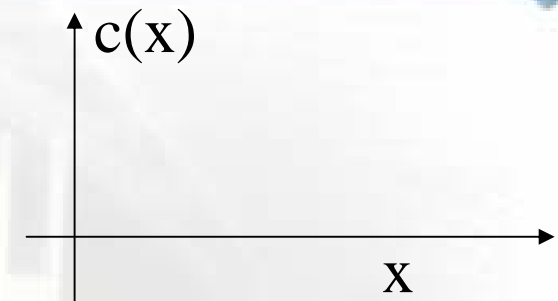




# CDF

Menampilkan fungsi kepadatan kumulatif (cumulative density function) dari bilangan acak

$$c(x_k) = \sum_{i=1}^k p(x_i)$$



```
>> x=floor(10*rand(1,16))
```

```
x =
```

```
Columns 1 through 6
```

```
6 7 6 0 4 4
```

```
Columns 7 through 12
```

```
3 1 6 6 7 4
```

```
Columns 13 through 16
```

```
5 1 4 7
```

```
>> t=0:9;
```

```
>> h=hist(x,t);
```

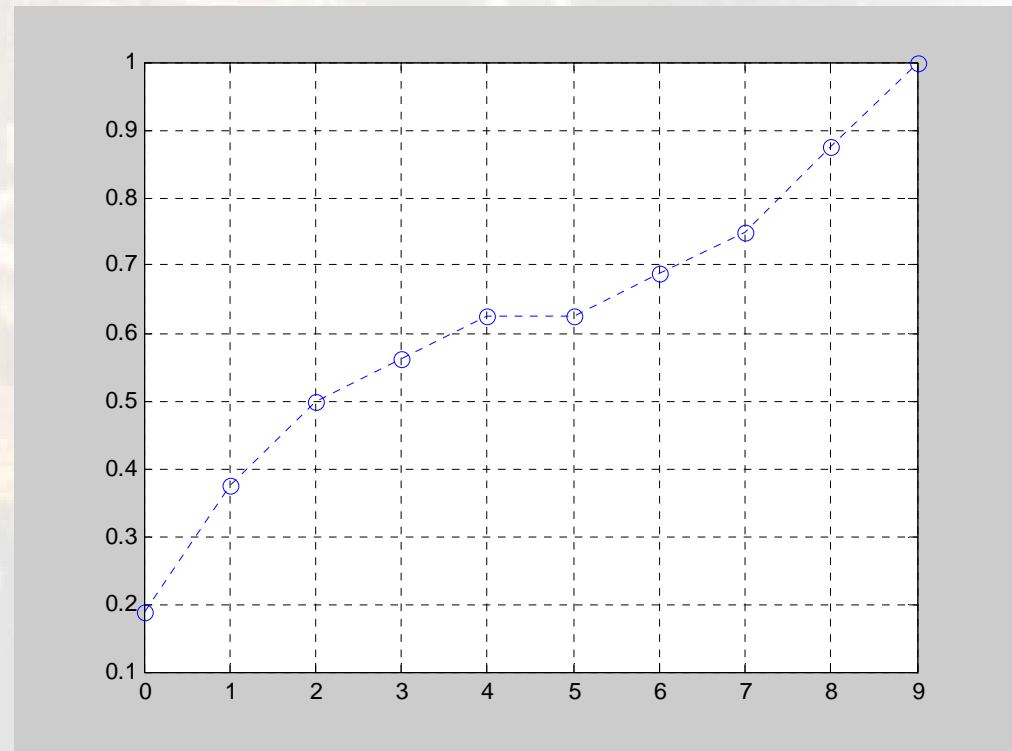
```
>> p=h/sum(h);
```

```
>> for i=0:9
```

```
c(i+1)=sum(p(1:i+1));
```

```
end
```

```
>> plot(t,c,'o'),grid
```



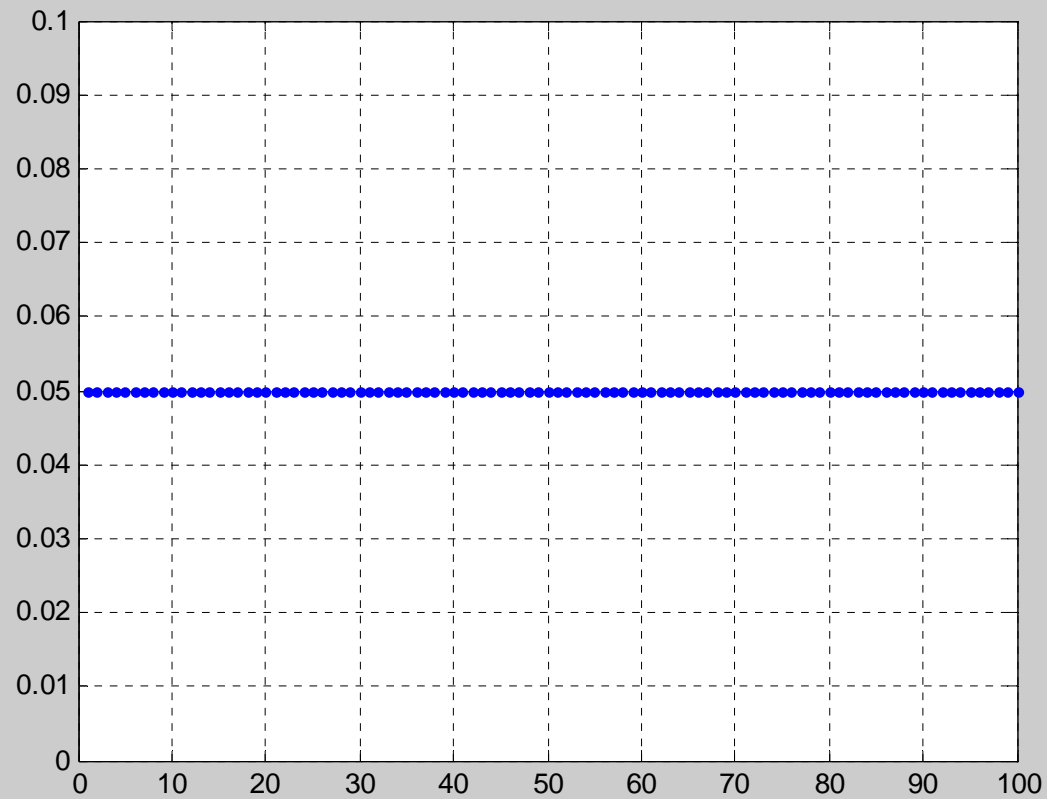


# Analisa Data Pada Bilangan Acak

- Perintah `rand()` menghasilkan bilangan acak dengan distribusi data uniform, artinya bahwa peluang munculnya setiap bilangan sama atau bila dilihat secara grafis, pdfnya berupa garis horisontal.
- Untuk menunjukkan distribusi bilangan acak yang dibangkitkan dapat digunakan nilai histogram, dan fungsi kepadatan probabilitas.
- Disarankan menggunakan pengujian distribusi untuk melakukan analisa terhadap distribusi dari bilangan acak yang dibangkitkan.



# Distribusi Uniform





## Tugas 2

Bangkitkan 20, 100, 500, 1000, 5000 dan 10000 bilangan acak bulat 0 s/d 9, kemudian gambarkan pdfnya. Perhatikan apakah ke-empat pembangkitan di atas menunjukkan distribusi uniform ?