

LAPORAN TUGAS TAMBAHAN

PROBABILITAS DAN STATISTIKA ET2101

Dosen Pengampu: Mohammad Sigit Arifianto, S.T, M.Sc.



Disusun oleh

Arya Bhagaskara

18123008

Al Rafi Fazle Mawla W

18123041

Christian VL Sebayang

18123040

DAFTAR ISI

DAFTAR ISI.....	2
BAB I.....	3
1.1 Distribusi Gaussian (Normal).....	3
1.2. Distribusi Eksponensial.....	3
1.3 Distribusi Uniform.....	4
BAB II.....	5
2.1. Distribusi Normal.....	5
2.1.1. Flowchart.....	5
2.1.2. Kode.....	6
2.1.3. Penjelasan Kode.....	7
2.1.4. Perbandingan dengan fungsi default.....	9
2.2. Distribusi Eksponensial.....	10
2.2.1. Flowchart.....	10
2.2.2. Kode.....	11
2.2.3. Penjelasan Kode.....	12
2.2.4. Perbandingan dengan fungsi default.....	14
2.3. Distribusi Uniform.....	15
2.3.1. Flowchart.....	15
2.3.2. Kode.....	16
2.3.3. Penjelasan Kode.....	17
2.3.4. Perbandingan dengan fungsi default.....	19

BAB I

DESKRIPSI DISTRIBUSI

1.1 Distribusi Gaussian (Normal)

Distribusi Gaussian, atau sering disebut distribusi normal, adalah salah satu distribusi probabilitas yang paling penting dalam statistik dan sains data. Distribusi ini memiliki bentuk simetris seperti lonceng, dengan rata-rata (μ) berada di tengah dan standar deviasi (σ) menentukan lebar kurva. Sebagian besar data dalam distribusi Gaussian berada di dekat rata-rata, dengan probabilitas menurun secara eksponensial saat bergerak menjauh dari rata-rata. Distribusi Gaussian digunakan untuk memodelkan fenomena alami seperti tinggi badan, berat badan, dan berbagai data lainnya.

Distribusi ini memiliki sifat penting yang dikenal sebagai **teorema limit pusat**, yang menyatakan bahwa ketika sejumlah besar data independen ditambahkan, hasil akhirnya akan mendekati distribusi Gaussian, terlepas dari distribusi awal data. Oleh karena itu, distribusi Gaussian sering digunakan sebagai asumsi dasar dalam analisis statistik. Properti penting lainnya adalah simetri, di mana data di sebelah kiri dan kanan rata-rata adalah cerminan satu sama lain.

Dalam aplikasi praktis, distribusi Gaussian digunakan dalam berbagai bidang, termasuk pemrosesan sinyal, pembelajaran mesin, dan analisis keuangan. Karena sifat matematisnya yang sederhana dan universal, distribusi ini sering digunakan untuk membuat model prediksi, analisis kesalahan, dan memahami pola data yang memiliki perilaku serupa dengan distribusi normal.

1.2. Distribusi Eksponensial

Distribusi eksponensial adalah distribusi probabilitas kontinu yang digunakan untuk memodelkan waktu antara peristiwa dalam proses yang terjadi secara kontinu dan independen pada tingkat rata-rata tertentu (λ). Salah satu contoh klasiknya adalah memodelkan waktu tunggu hingga sebuah telepon berdering di call center atau waktu antara kedatangan pelanggan di restoran. Parameter laju (λ) menentukan seberapa cepat peristiwa terjadi; semakin besar λ , semakin pendek waktu antara peristiwa.

Karakteristik utama dari distribusi eksponensial adalah sifat "**tanpa ingatan**", yang berarti probabilitas kejadian di masa depan tidak dipengaruhi oleh waktu yang telah berlalu. Ini membuat distribusi eksponensial sangat berguna untuk memodelkan kejadian acak seperti keandalan sistem, waktu kegagalan perangkat, atau interaksi partikel dalam fisika.

Dalam dunia nyata, distribusi eksponensial sering digunakan dalam teori antrean, analisis keandalan, dan analisis risiko. Sifat eksponensial memungkinkan perhitungan probabilitas yang mudah, sehingga mempermudah simulasi dan analisis untuk sistem di mana waktu antar kejadian adalah faktor kunci.

1.3 Distribusi Uniform

Distribusi uniform adalah distribusi probabilitas di mana semua nilai dalam rentang tertentu memiliki kemungkinan yang sama. Distribusi ini didefinisikan oleh dua parameter, yaitu batas bawah (aaa) dan

batas atas (bbb), dengan probabilitas tetap di seluruh interval tersebut. Salah satu contoh nyata adalah pelemparan dadu, di mana setiap sisi memiliki peluang yang sama untuk muncul.

Distribusi uniform memiliki bentuk sederhana dan sering digunakan sebagai dasar untuk membangkitkan bilangan acak dalam banyak aplikasi. Dalam simulasi Monte Carlo, distribusi uniform digunakan untuk mensimulasikan berbagai skenario karena sifatnya yang tidak bias. Distribusi ini juga digunakan dalam algoritma sampling untuk menghasilkan bilangan acak yang diperlukan untuk eksperimen.

Dalam aplikasi praktis, distribusi uniform digunakan untuk memodelkan kejadian di mana semua hasil memiliki peluang yang sama. Ini mencakup bidang seperti pengujian perangkat lunak, analisis probabilitas dalam permainan, dan pemodelan situasi di mana tidak ada preferensi khusus terhadap nilai tertentu dalam suatu interval. Sifat kesederhanaan dan fleksibilitasnya menjadikan distribusi uniform sebagai alat yang sangat berguna dalam analisis data.

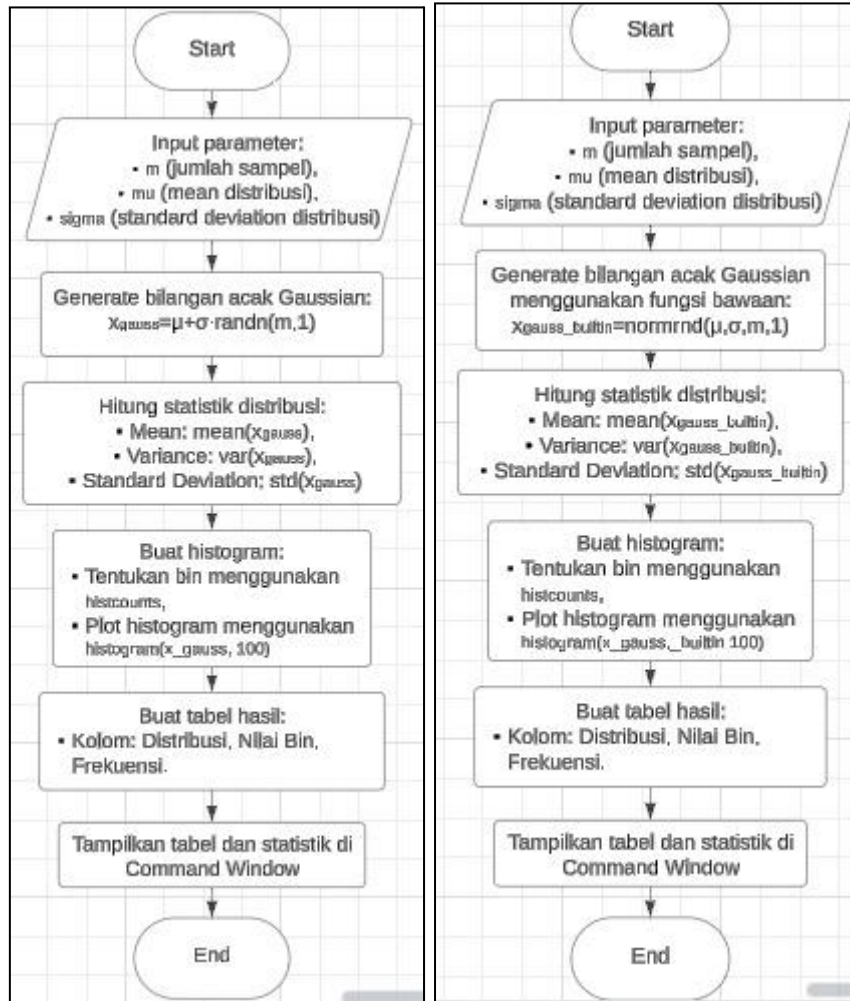
BAB II

IMPLEMENTASI DISTRIBUSI KONTINU

2.1. Distribusi Normal

2.1.1. Flowchart

Gambar 1: Flowchart Distribusi Normal



2.1.2. Kode

```
function gaussian_comparison(m, mu, sigma)
% gaussian_comparison.m
% Compare custom Gaussian function with normrnd (Statistics and Machine Learning Toolbox)
% Inputs:
% m - Number of samples
% mu - Mean of the Gaussian distribution
% sigma - Standard deviation of the Gaussian distribution

% Custom Gaussian Distribution
x_gauss_custom = mu + (sigma * randn(m, 1));

% Built-in Gaussian Distribution (normrnd)
x_gauss_builtin = normrnd(mu, sigma, m, 1);

% Calculate statistics for both distributions
mu_gauss_custom = mean(x_gauss_custom);
sigma_gauss_custom = std(x_gauss_custom);

mu_gauss_builtin = mean(x_gauss_builtin);
sigma_gauss_builtin = std(x_gauss_builtin);

% Plot histograms for comparison
figure;
subplot(1, 2, 1);
histogram(x_gauss_custom, 100);
title('Custom Gaussian Distribution');
xlabel('Value');
ylabel('Frequency');

subplot(1, 2, 2);
histogram(x_gauss_builtin, 100);
title('Built-in Gaussian Distribution (normrnd)');
xlabel('Value');
ylabel('Frequency');

% Calculate frequencies for histogram bins
[counts_custom, edges_custom] = histcounts(x_gauss_custom, 100);
[counts_builtin, edges_builtin] = histcounts(x_gauss_builtin, 100);

% Create comparison table
Distribution = [repmat({'Custom'}, length(edges_custom)-1, 1); ...
               repmat({'Built-in'}, length(edges_builtin)-1, 1)];
Bin_Values = [edges_custom(1:end-1); edges_builtin(1:end-1)];
Frequency = [counts_custom'; counts_builtin'];

results_table = table(Distribution, Bin_Values, Frequency);

% Display the table in the Command Window
disp(results_table);

% Display statistical comparison in Command Window
fprintf('\nGaussian Distribution Comparison:\n');
fprintf('Custom: Mean = %.2f, Std = %.2f\n', mu_gauss_custom, sigma_gauss_custom);
fprintf('Built-in (normrnd): Mean = %.2f, Std = %.2f\n', mu_gauss_builtin, sigma_gauss_builtin);

end
```

2.1.3. Penjelasan Kode

1. Menghasilkan Data Gaussian

Custom Gaussian:

```
x_gauss_custom = mu + (sigma * randn(m, 1));
```

Data dihasilkan dengan menambahkan rata-rata mu ke nilai acak dari distribusi normal standar (randn) yang telah disesuaikan dengan standar deviasi sigma.

Built-in Gaussian:

```
x_gauss_builtin = normrnd(mu, sigma, m, 1);
```

Menggunakan fungsi normrnd untuk menghasilkan data dengan rata-rata mu dan standar deviasi sigma.

2. Menghitung Statistik

Custom Distribution:

```
mu_gauss_custom = mean(x_gauss_custom);
```

```
sigma_gauss_custom = std(x_gauss_custom);
```

Menghitung rata-rata dan standar deviasi dari data distribusi Gaussian manual.

Built-in Distribution:

```
mu_gauss_builtin = mean(x_gauss_builtin);
```

```
sigma_gauss_builtin = std(x_gauss_builtin);
```

Menghitung rata-rata dan standar deviasi dari data distribusi Gaussian bawaan.

3. Visualisasi Data

Membuat histogram dari kedua dataset untuk membandingkan distribusi:

```
subplot(1, 2, 1);
```

```
histogram(x_gauss_custom, 100);
```

```
title('Custom Gaussian Distribution');
```

```
subplot(1, 2, 2);
```

```
histogram(x_gauss_builtin, 100);
```

```
title('Built-in Gaussian Distribution (normrnd)');
```

Histogram dibuat dengan 100 bin.

4. Menghitung Frekuensi untuk Histogram

Custom Distribution:

```
[counts_custom, edges_custom] = histcounts(x_gauss_custom, 100);
```

Menghitung frekuensi tiap bin untuk data Gaussian manual.

Built-in Distribution:

```
[counts_builtin, edges_builtin] = histcounts(x_gauss_builtin, 100);
```

Menghitung frekuensi tiap bin untuk data Gaussian bawaan.

5. Membuat Tabel Perbandingan

Membuat tabel untuk membandingkan distribusi custom dan bawaan:

```
results_table = table(Distribution, Bin_Values, Frequency);
```

```
disp(results_table);
```

Tabel berisi informasi:

Distribution: Jenis distribusi (Custom atau Built-in).

Bin_Values: Nilai awal tiap bin.

Frequency: Jumlah sampel dalam tiap bin.

6. Menampilkan Statistik di Command Window

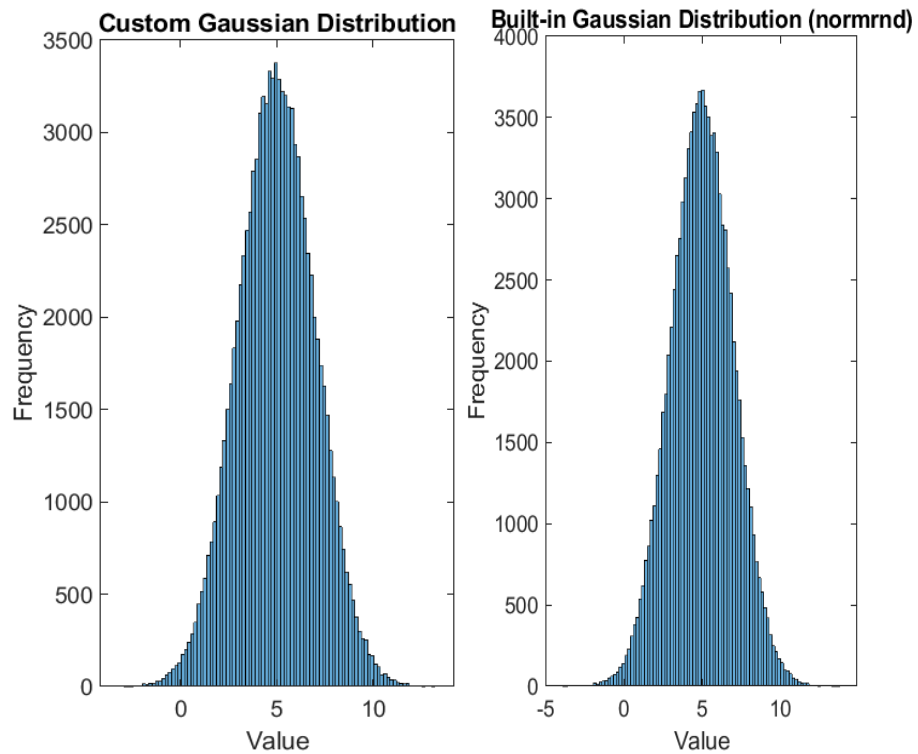
Menampilkan perbandingan statistik kedua distribusi:

```
fprintf('Custom: Mean = %.2f, Std = %.2f\n', mu_gauss_custom, sigma_gauss_custom);
```

```
fprintf('Built-in (normrnd): Mean = %.2f, Std = %.2f\n', mu_gauss_builtin, sigma_gauss_builtin);
```


2.1.4. Perbandingan dengan fungsi default

Gambar 2: Perbandingan Grafik Gaussian



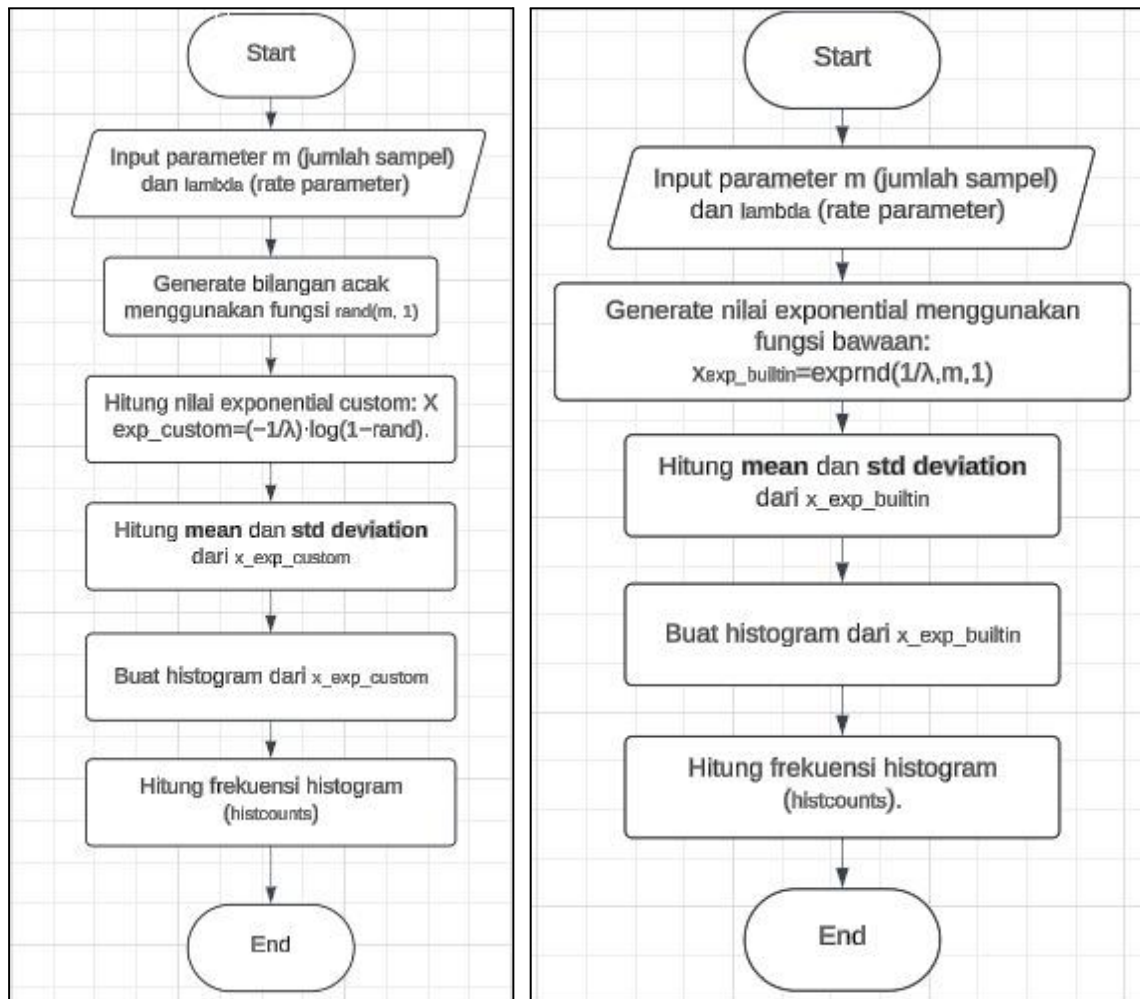
Distribution	Bin_Values	Frequency
Custom	-33	1
Custom	-32	2
Custom	-31	3
Custom	-30	4
Custom	-29	5
Custom	-28	6
Custom	-27	7
Custom	-26	8
Custom	-25	9
Custom	-24	10
Custom	-23	11
Custom	-22	12
Custom	-21	13

Distribution	Bin_Values	Frequency
Built-in	-33	1
Built-in	-32	2
Built-in	-31	3
Built-in	-30	4
Built-in	-29	5
Built-in	-28	6
Built-in	-27	7
Built-in	-26	8
Built-in	-25	9
Built-in	-24	10
Built-in	-23	11
Built-in	-22	12
Built-in	-21	13

2.2. Distribusi Eksponensial

2.2.1. Flowchart

Gambar 3: Flowchart Distribusi Eksponensial



2.2.2. Kode

```
function exponential_distribution(m, lambda)
% exponential_distribution.m
% This function generates Exponential random samples and compares
% them with MATLAB's built-in exprnd function. (compare define function with machine_learning toolbox)
%
% Inputs:
% m - Number of samples
% lambda - Rate parameter for the Exponential distribution

% Custom Exponential Distribution
x_exp_custom = - (1 / lambda) * log(1 - rand(m, 1));

% Built-in Exponential Distribution (exprnd)
x_exp_builtin = exprnd(1 / lambda, m, 1);

% Calculate statistics for both distributions
mu_exp_custom = mean(x_exp_custom);
sigma_exp_custom = std(x_exp_custom);

mu_exp_builtin = mean(x_exp_builtin);
sigma_exp_builtin = std(x_exp_builtin);

% Plot histograms for comparison
figure;
subplot(1, 2, 1);
histogram(x_exp_custom, 100);
title('Custom Exponential Distribution');
xlabel('Value');
ylabel('Frequency');

subplot(1, 2, 2);
histogram(x_exp_builtin, 100);
title('Built-in Exponential Distribution (exprnd)');
xlabel('Value');
ylabel('Frequency');

% Calculate frequencies for histogram bins
[counts_custom, edges_custom] = histcounts(x_exp_custom, 100);
[counts_builtin, edges_builtin] = histcounts(x_exp_builtin, 100);

% Create comparison table
Distribution = [repmat({'Custom'}, length(edges_custom)-1, 1); ...
               repmat({'Built-in'}, length(edges_builtin)-1, 1)];
Bin_Values = [edges_custom(1:end-1); edges_builtin(1:end-1)];
Frequency = [counts_custom'; counts_builtin'];

results_table = table(Distribution, Bin_Values, Frequency);

% Display the table in the Command Window
disp(results_table);

% Display statistical comparison in Command Window
fprintf("\nExponential Distribution Comparison:\n");
fprintf('Custom: Mean = %.2f, Std = %.2f\n', mu_exp_custom, sigma_exp_custom);
fprintf('Built-in (exprnd): Mean = %.2f, Std = %.2f\n', mu_exp_builtin, sigma_exp_builtin);
end
```

2.2.3. Penjelasan Kode

1. Menghasilkan Data Eksponensial

Custom Exponential:

```
x_exp_custom = -(1 / lambda) * log(1 - rand(m, 1));
```

Data dihasilkan menggunakan rumus distribusi eksponensial, dimana U adalah bilangan acak dari distribusi uniform dalam interval (0, 1) yang dihasilkan oleh rand.

Built-in Exponential:

```
x_exp_builtin = exprnd(1 / lambda, m, 1);
```

Fungsi exprnd menghasilkan data eksponensial dengan parameter rata-rata

2. Menghitung Statistik

Custom Distribution:

```
mu_exp_custom = mean(x_exp_custom);  
sigma_exp_custom = std(x_exp_custom);
```

Menghitung rata-rata (mean) dan standar deviasi (std) dari data distribusi eksponensial manual.

Built-in Distribution:

```
mu_exp_builtin = mean(x_exp_builtin);  
sigma_exp_builtin = std(x_exp_builtin);
```

Menghitung rata-rata dan standar deviasi dari data distribusi eksponensial bawaan.

3. Visualisasi Data

Membuat histogram dari kedua dataset untuk membandingkan distribusi:

```
subplot(1, 2, 1);  
histogram(x_exp_custom, 100);  
title('Custom Exponential Distribution');  
  
subplot(1, 2, 2);  
histogram(x_exp_builtin, 100);  
title('Built-in Exponential Distribution (exprnd)');
```

Histogram dibuat dengan 100 bin.

4. Menghitung Frekuensi Histogram

Custom Distribution:

```
[counts_custom, edges_custom] = histcounts(x_exp_custom, 100);
```

Menghitung frekuensi tiap bin untuk data distribusi eksponensial manual.

Built-in Distribution:

```
[counts_builtin, edges_builtin] = histcounts(x_exp_builtin, 100);
```

Menghitung frekuensi tiap bin untuk data distribusi eksponensial bawaan.

5. Membuat Tabel Perbandingan

Membuat tabel berisi:

Distribution: Menyebutkan apakah data berasal dari metode manual atau bawaan.

Bin_Values: Nilai awal tiap bin pada histogram.

Frequency: Frekuensi data pada tiap bin.

```
results_table = table(Distribution, Bin_Values, Frequency);  
disp(results_table);
```

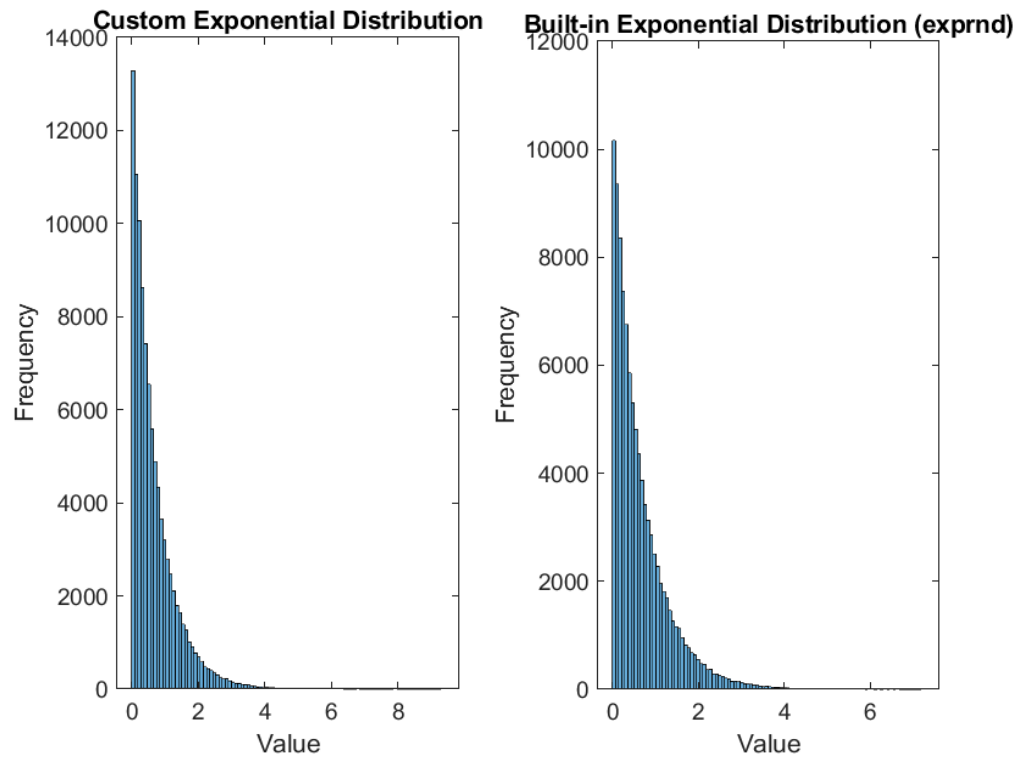
6. Menampilkan Statistik di Command Window

Menampilkan statistik rata-rata dan standar deviasi dari kedua metode:

```
fprintf('Custom: Mean = %.2f, Std = %.2f\n', mu_exp_custom, sigma_exp_custom);  
fprintf('Built-in (exprnd): Mean = %.2f, Std = %.2f\n\n', mu_exp_builtin, sigma_exp_builtin);
```

2.2.4. Perbandingan dengan fungsi default

Gambar 4: Perbandingan Grafik Eksponensial



Distribution	Bin_Values	Frequency
Custom	0	13276
Custom	0,0937	11064
Custom	0,1874	10062
Custom	0,2811	8630
Custom	0,3748	7419
Custom	0,4685	6547
Custom	0,5622	5590
Custom	0,6559	4892
Custom	0,7496	4340
Custom	0,8433	3643

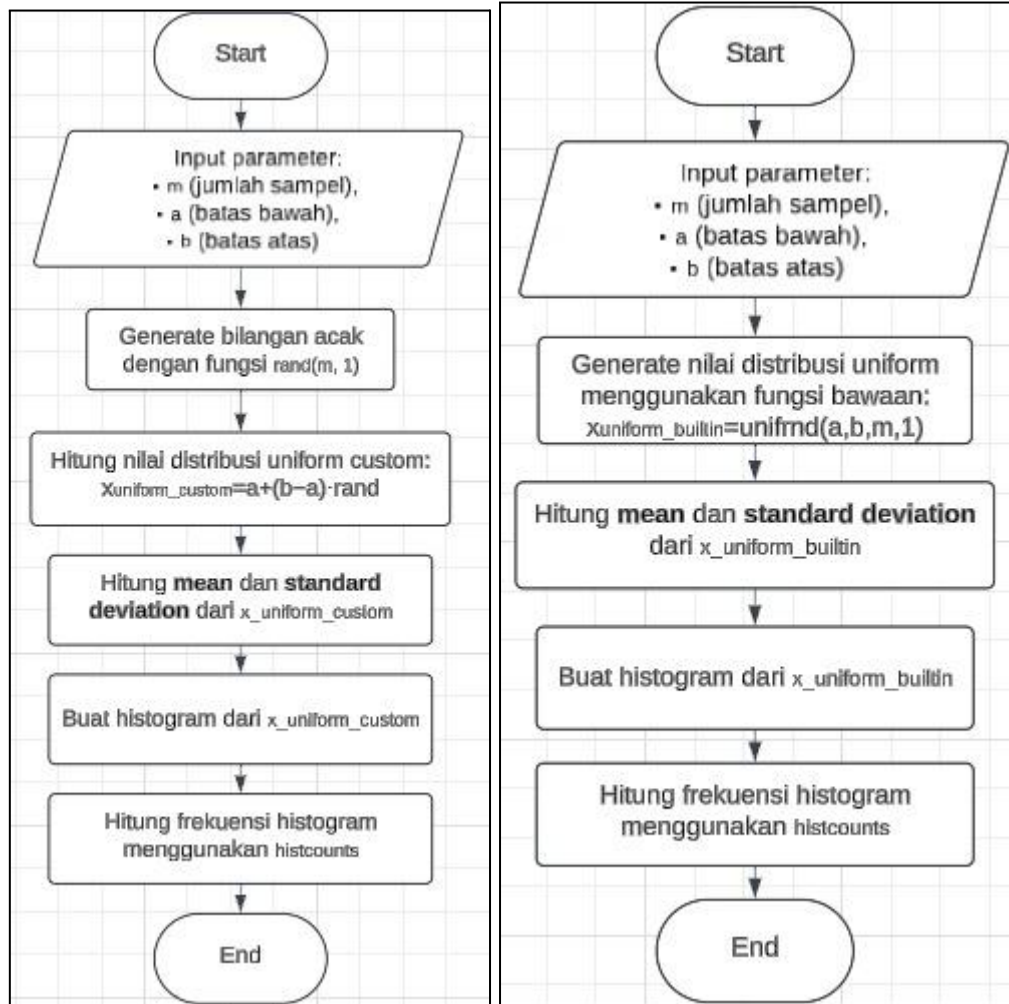
Distribution	Bin_Values	Frequency
Built-in	0	10161
Built-in	0,0724	9354
Built-in	0,1448	8352
Built-in	0,2172	7378
Built-in	0,2896	6752
Built-in	0,362	5857
Built-in	0,4344	5307
Built-in	0,5068	4818
Built-in	0,5792	4361
Built-in	0,6516	3873

Tabel 2: Perbandingan Value Distribusi Eksponensial

2.3. Distribusi Uniform

2.3.1. Flowchart

Gambar 5: Flowchart Distribusi Uniform



2.3.2. Kode

```
function uniform_distribution(m, a, b)
% uniform_distribution.m
% This function generates Uniform random samples and compares
% them with MATLAB's built-in unifrnd function. (compare with the task statics function and machine learnig toolbox)
%
% Inputs:
% m - Number of samples
% a - Lower bound of the Uniform distribution
% b - Upper bound of the Uniform distribution

% Custom Uniform Distribution
x_uniform_custom = a + (b - a) * rand(m, 1);

% Built-in Uniform Distribution (unifrnd)
x_uniform_builtin = unifrnd(a, b, m, 1);

% Calculate statistics for both distributions
mu_uniform_custom = mean(x_uniform_custom);
sigma_uniform_custom = std(x_uniform_custom);

mu_uniform_builtin = mean(x_uniform_builtin);
sigma_uniform_builtin = std(x_uniform_builtin);

% Plot histograms for comparison
figure;
subplot(1, 2, 1);
histogram(x_uniform_custom, 100);
title('Custom Uniform Distribution');
xlabel('Value');
ylabel('Frequency');

subplot(1, 2, 2);
histogram(x_uniform_builtin, 100);
title('Built-in Uniform Distribution (unifrnd)');
xlabel('Value');
ylabel('Frequency');

% Calculate frequencies for histogram bins
[counts_custom, edges_custom] = histcounts(x_uniform_custom, 100);
[counts_builtin, edges_builtin] = histcounts(x_uniform_builtin, 100);

% Create comparison table
Distribution = [repmat('Custom', length(edges_custom)-1, 1); ...
               repmat('Built-in', length(edges_builtin)-1, 1)];
Bin_Values = [edges_custom(1:end-1); edges_builtin(1:end-1)];
Frequency = [counts_custom; counts_builtin];

results_table = table(Distribution, Bin_Values, Frequency);

% Display the table in the Command Window
disp(results_table);

% Display statistical comparison in Command Window
fprintf('\nUniform Distribution Comparison:\n');
fprintf('Custom: Mean = %.2f, Std = %.2f\n', mu_uniform_custom, sigma_uniform_custom);
fprintf('Built-in (unifrnd): Mean = %.2f, Std = %.2f\n', mu_uniform_builtin, sigma_uniform_builtin);
end
```


2.3.3. Penjelasan Kode

1. Menghasilkan Data Uniform

Custom Uniform:

```
x_uniform_custom = a + (b - a) * rand(m, 1);
```

Data dihasilkan dengan menyesuaikan bilangan acak uniform standar $U \sim [0,1]$ (dihasilkan oleh rand) agar berada pada interval $[a,b]$

Built-in Uniform:

```
x_uniform_builtin = unifrnd(a, b, m, 1);
```

Fungsi unifrnd langsung menghasilkan data uniform dalam interval $[a,b]$

2. Menghitung Statistik

Custom Distribution:

```
mu_uniform_custom = mean(x_uniform_custom);  
sigma_uniform_custom = std(x_uniform_custom);
```

Menghitung rata-rata (mean) dan standar deviasi (std) dari data distribusi uniform manual.

Built-in Distribution:

```
mu_uniform_builtin = mean(x_uniform_builtin);  
sigma_uniform_builtin = std(x_uniform_builtin);
```

Menghitung rata-rata dan standar deviasi dari data distribusi uniform bawaan.

3. Visualisasi Data

Membuat histogram dari kedua dataset untuk membandingkan distribusi:

```
subplot(1, 2, 1);  
histogram(x_uniform_custom, 100);  
title('Custom Uniform Distribution');  
  
subplot(1, 2, 2);  
histogram(x_uniform_builtin, 100);  
title('Built-in Uniform Distribution (unifrnd)');
```

Histogram dibuat dengan 100 bin.

4. Menghitung Frekuensi Histogram

Custom Distribution:

```
[counts_custom, edges_custom] = histcounts(x_uniform_custom, 100);
```

Menghitung jumlah sampel dalam tiap bin untuk data distribusi uniform manual.

Built-in Distribution:

```
[counts_builtin, edges_builtin] = histcounts(x_uniform_builtin, 100);
```

Menghitung jumlah sampel dalam tiap bin untuk data distribusi uniform bawaan.

5. Membuat Tabel Perbandingan

Membuat tabel berisi:

Distribution: Menyebutkan apakah data berasal dari metode manual atau bawaan.

Bin_Values: Nilai awal tiap bin pada histogram.

Frequency: Jumlah sampel dalam tiap bin.

```
results_table = table(Distribution, Bin_Values, Frequency);
disp(results_table);
```

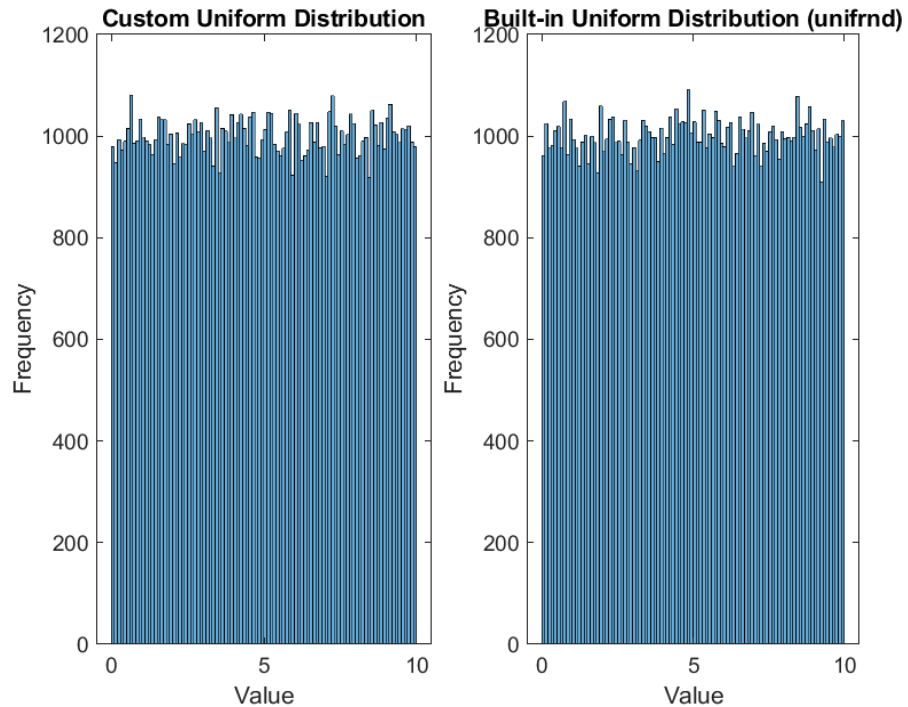
6. Menampilkan Statistik di Command Window

Menampilkan statistik rata-rata dan standar deviasi dari kedua metode:

```
fprintf('Custom:   Mean   =   %.2f,   Std   =   %.2f\n', mu_uniform_custom,
sigma_uniform_custom);
fprintf('Built-in (unifrnd): Mean = %.2f, Std = %.2f\n\n', mu_uniform_builtin,
sigma_uniform_builtin);
```

2.3.4. Perbandingan dengan fungsi default

Gambar 7: Perbandingan Grafik Distribusi Uniform



Distribution	Bin_Values	Frequency
Custom	0	979
Custom	0,1	948
Custom	0,2	993
Custom	0,3	973
Custom	0,4	990
Custom	0,5	1015
Custom	0,6	1081
Custom	0,7	986

Distribution	Bin_Values	Frequency
Built-in	0	962
Built-in	0,1	1025
Built-in	0,2	976
Built-in	0,3	982
Built-in	0,4	1011
Built-in	0,5	1019
Built-in	0,6	976
Built-in	0,7	1068

Tabel 3: Perbandingan Value Distribusi Uniform