

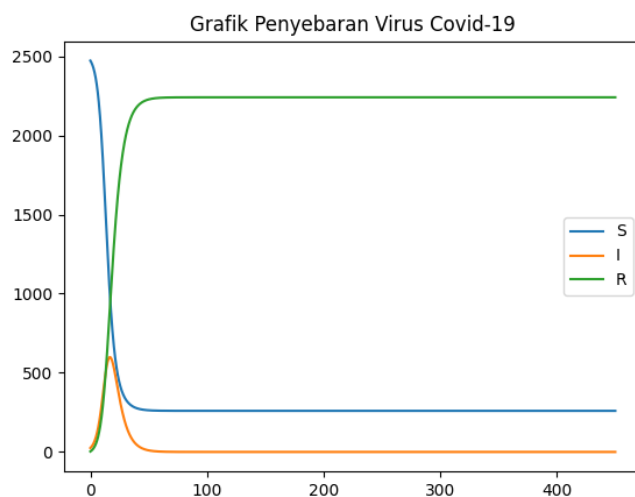
PRAKTIKUM FISIKA KOMPUTASI
MODUL 5
PENYELESAIAN MODEL SIR MENGGUNAKAN METODE UELER

Nama : Muhammad Raza Naufal

NIM : 1227030023

A. Analisis Grafik Penyebaran Virus Covid-19

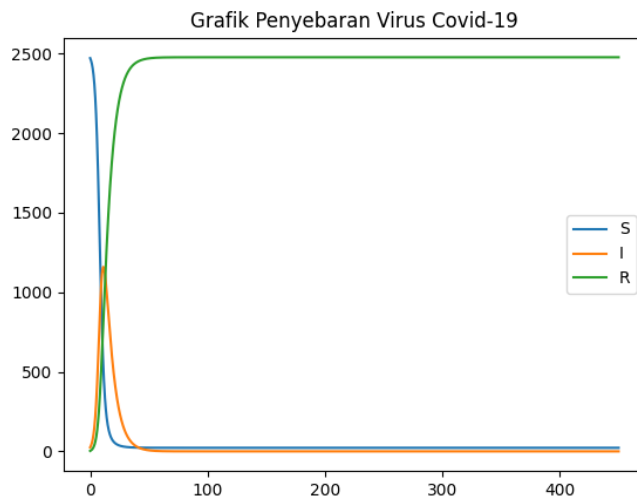
1. Grafik dengan laju penularan 0.5 dan laju pemulihan 0.2



Pada grafik pertama ini, kita dapat melihat bahwa jumlah populasi yang terinfeksi (**I**) naik di awal, namun kenaikan hanya dalam waktu yang singkat hingga mencapai puncaknya. Seiring dengan meningkatnya jumlah orang yang terinfeksi, jumlah orang yang rentan (**S**) menurun dengan cepat, menunjukkan bahwa virus menyebar dengan cukup

cepat. Setelah puncaknya, jumlah orang yang sembuh (**R**) terus meningkat secara stabil, dan pada akhirnya mendominasi populasi, sementara jumlah orang yang terinfeksi turun mendekati nol. Waktu yang dibutuhkan untuk mengendalikan penyebaran cukup cepat karena laju pemulihan yang relatif tinggi (0.2), meskipun laju penularan juga cukup tinggi (0.5).

2. Grafik dengan laju penularan 0.7 dan laju pemulihan 0.15



Pada grafik kedua ini, laju penularan lebih tinggi (0.7), yang menyebabkan jumlah orang yang terinfeksi naik lebih cepat dan lebih tajam dibandingkan grafik pertama. Jumlah orang yang rentan (**S**) menurun lebih cepat karena virus menyebar lebih cepat kepada populasi. Namun, karena laju pemulihan lebih rendah (0.15), orang yang terinfeksi

membutuhkan waktu lebih lama untuk sembuh, menyebabkan kurva infeksi bertahan lebih lama sebelum menurun. Grafik ini menunjukkan bahwa meskipun jumlah orang yang akhirnya sembuh (**R**) mendominasi seperti pada grafik pertama, proses penyembuhan berlangsung lebih lambat, dengan kurva infeksi yang lebih tinggi dan lebih lama.

B. Penjelasan Algoritma Pemrograman

```
import numpy as np
import matplotlib.pyplot as plt
```

Mengimport library yang digunakan yaitu numpy dan matplotlib untuk membuat grafik

```
t0 = 0
tn = 450
ndata = 2500

t = np.linspace(t0, tn, ndata)
h = t[2]-t[1]
```

t_0 dan t_n adalah batas waktu simulasi, dimana t_0 merupakan waktu awal yaitu 0, sedangkan t_n adalah waktu dalam 450 hari seperti yang diperintahkan dalam soal. Kemudian $ndata$ merupakan sama dengan jumlah populasi yaitu 2500 sesuai dengan perintah. Lalu t adalah array yang berisi titik-titik waktu yang dihasilkan dari fungsi `np.linspace()`, yang membagi interval waktu dari t_0 hingga t_n menjadi $ndata$ bagian yang sama. Sedangkan h adalah langkah waktu atau interval antar titik dalam array waktu. Ini didapat dari selisih antara dua elemen berturut-turut dalam array t .

```

N = 2500
I0 = 25
R0 = 3
S0 = N - I0 - R0

I = np.zeros(ndata)
S = np.zeros(ndata)
R = np.zeros(ndata)

I[0] = I0
S[0] = S0
R[0] = R0

```

N adalah jumlah populasi, disamakan dengan jumlah data yaitu 2500 sesuai dengan soal. I0 adalah jumlah awal orang yang terinfeksi yaitu 25, kemudian R0 adalah jumlah awal orang yang sembuh yaitu 3 orang, dan S0 merupakan jumlah orang yang rentan, dihitung dengan rumus $N - I_0 - R_0$. Array I, S, dan R digunakan untuk menyimpan jumlah orang yang terinfeksi (I), yang rentan (S), dan yang sembuh (R) pada setiap titik waktu simulasi. Pada titik awal ($n=0$), kondisi awal masing-masing variabel adalah $I[0] = I_0$, $S[0] = S_0$, dan $R[0] = R_0$.

```

beta = 0.5
gamma = 0.2

```

beta adalah laju penularan yang menentukan seberapa cepat virus menyebar. Sedangkan gamma adalah laju pemulihan, yang menentukan seberapa cepat orang yang terinfeksi sembuh.

```

for n in range(0, ndata-1):
    S[n+1] = S[n] - h*beta/N*S[n]*I[n]
    I[n+1] = I[n] + h*beta/N*S[n]*I[n] - h*gamma*I[n]
    R[n+1] = R[n] + h*gamma*I[n]

```

Fungsi ini menghitung dinamika S, I, dan R dari waktu ke waktu menggunakan metode Euler. Dimana $S[n+1]$ menghitung jumlah populasi yang rentan pada waktu berikutnya berdasarkan penurunan jumlah orang yang terinfeksi. Perubahan ini dipengaruhi oleh beta, N, $S[n]$, dan $I[n]$ (laju infeksi). Kemudian $I[n+1]$ menghitung jumlah populasi yang terinfeksi pada waktu berikutnya berdasarkan peningkatan jumlah orang yang rentan yang terinfeksi dan pengurangan karena kesembuhan (gamma). Lalu $R[n+1]$ menghitung jumlah populasi yang sembuh pada waktu berikutnya berdasarkan jumlah orang yang sembuh dari kelompok terinfeksi.

```

plt.plot(t, S, label='S')
plt.plot(t, I, label='I')
plt.plot(t, R, label='R')
plt.title('Grafik Penyebaran Virus Covid-19')
plt.legend()
plt.show()

```

Fungsi ini untuk memvisualisasi data dalam bentuk grafik, menunjukkan dinamika penyebaran virus di dalam populasi dengan label untuk masing-masing kurva. untuk S sebagai orang yang rentan, kemudian I sebagai orang yang terinfeksi dan R sebagai orang yang sembuh.