PRAKTIKUM FISIKA KOMPUTASI

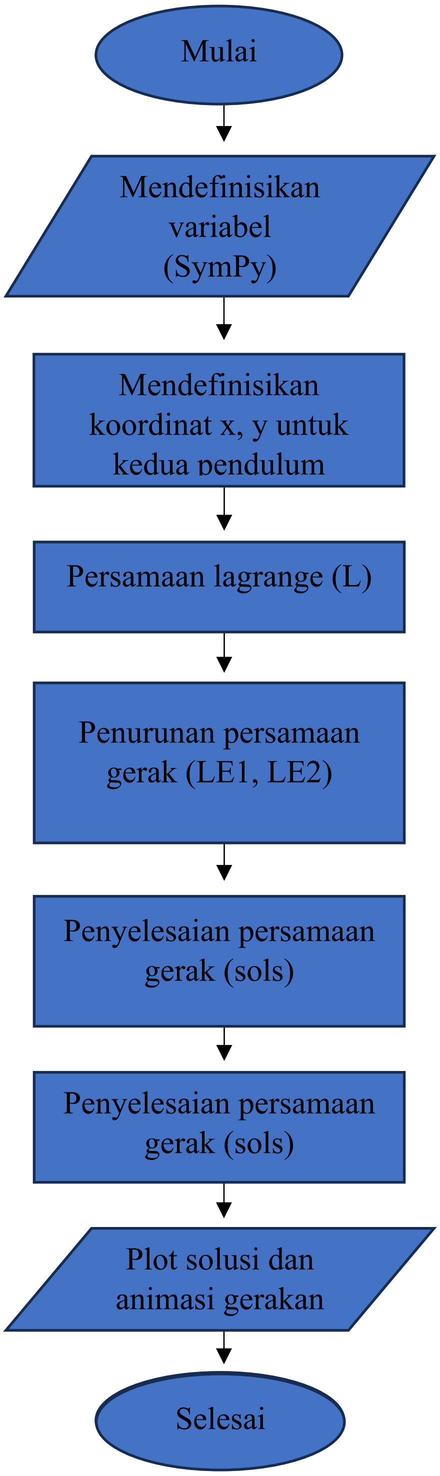
MODUL 8

ANALISIS DOUBLE PENDULUM

Nama : Muhammad Raza Naufal

NIM : 1227030023

- ***Flowchart* kode pemrograman double pendulum**



- **Algoritma Kode program**

1. **Inisialisasi Variabel dan Parameter**
   Program dimulai dengan mendefinisikan variabel-variabel simbolik yang digunakan untuk perhitungan fisika pendulum ganda. Variabel ini termasuk waktu, panjang tali, massa, gravitasi, dan sudut pendulum beserta turunan pertama dan keduanya. Pendekatan ini menggunakan library **SymPy** untuk memungkinkan manipulasi simbolik yang fleksibel. Dengan mendefinisikan turunan sudut terhadap waktu, persamaan gerak nantinya dapat diperoleh melalui kalkulus diferensial.

```
t, m, g, L1, L2, w, C, alph, beta = smp.symbols(r't m g L_1, L_2 \omega C \alpha \beta')

the1, the2, = smp.symbols(r'\theta_1 \theta_2 ', cls=smp.Function)

the1 = the1(t)
the1_d = smp.diff(the1, t)
the1_dd = smp.diff(the1_d, t)

the2 = the2(t)
the2_d = smp.diff(the2, t)
the2_dd = smp.diff(smp.diff(the2, t), t)
```

2. **Menghitung Koordinat dan Kecepatan**
   Koordinat masing-masing pendulum dihitung menggunakan sudut rotasi dan panjang tali. Pendulum pertama memiliki posisi relatif terhadap titik asal, sedangkan pendulum kedua bergantung pada posisi pendulum pertama. Dengan menggunakan turunan terhadap waktu, kecepatan masing-masing pendulum juga dihitung. Posisi ini penting untuk menentukan energi kinetik sistem.

```
x1, y1, x2, y2 = smp.symbols('x_1, y_1, x_2, y_2', cls=smp.Function)
x1 = x1(t, the1)
y1 = y1(t, the1)
x2 = x2(t, the1, the2)
y2 = y2(t, the1, the2)

x1 = smp.cos(w*t) + L1 * smp.sin(the1)
y1 = -L1 * smp.cos(the1)
x2 = smp.cos(w*t) + L1 * smp.sin(the1) + L2 * smp.sin(the2)
y2 = -L1 * smp.cos(the1) - L2 * smp.cos(the2)
```

3. **Menyusun Lagrangian**
   Energi kinetik dihitung dari kecepatan kuadrat masing-masing pendulum, sementara energi potensial dihitung berdasarkan posisi vertikal relatif terhadap gravitasi. Lagrangian didefinisikan sebagai selisih antara energi kinetik dan potensial. Fungsi ini menjadi dasar untuk mendeskripsikan dinamika sistem menggunakan prinsip Euler-Lagrange.

```
T = 1/2 * (smp.diff(x1, t)**2 + smp.diff(y1, t)**2) + \
    1/2 * m * (smp.diff(x2, t)**2 + + smp.diff(y2, t)**2)
V = g * y1 + m * g * y2
L = T - V
```

4. **Menurunkan Persamaan Gerak**
   Dengan menerapkan persamaan Euler-Lagrange, persamaan gerak untuk masing-masing sudut theta diperoleh. Persamaan ini melibatkan turunan Lagrangian terhadap sudut serta turunan waktu dari turunan parsial terhadap kecepatan sudut. Hasilnya adalah dua persamaan diferensial yang menggambarkan dinamika sistem.

```
# Persamaan Lagrange-Euler untuk theta1
LE1 = smp.diff(L, the1) - smp.diff(smp.diff(L, the1_d), t)
LE1 = LE1.simplify()
```

```
# Persamaan Lagrange-Euler untuk theta2
LE2 = smp.diff(L, the2) - smp.diff(smp.diff(L, the2_d), t)
LE2 = LE2.simplify()
```

5. **Menyelesaikan Persamaan Gerak**
   Persamaan gerak diferensial yang dihasilkan dari langkah sebelumnya diselesaikan untuk memperoleh percepatan sudut. Solusi ini dalam bentuk simbolik dan nantinya akan dikonversi ke fungsi numerik untuk keperluan simulasi.

```
sols = smp.solve([LE1, LE2], (the1_dd, the2_dd),
                 simplify=False, rational=False)
```

6. **Konversi ke Fungsi Numerik**
   Solusi simbolik yang diperoleh kemudian diubah menjadi fungsi Python menggunakan **SymPy**. Fungsi ini mempermudah pemakaian hasil perhitungan simbolik dalam integrasi numerik berikutnya.

```
dz1dt_f = smp.lambdify((t, m, g, w, L1, L2, the1, the2, the1_d, the2_d), sols[the1_dd])
dthe1dt_f = smp.lambdify(the1_d, the1_d)

dz2dt_f = smp.lambdify((t, m, g, w, L1, L2, the1, the2, the1_d, the2_d), sols[the2_dd])
dthe2dt_f = smp.lambdify(the2_d, the2_d)
```

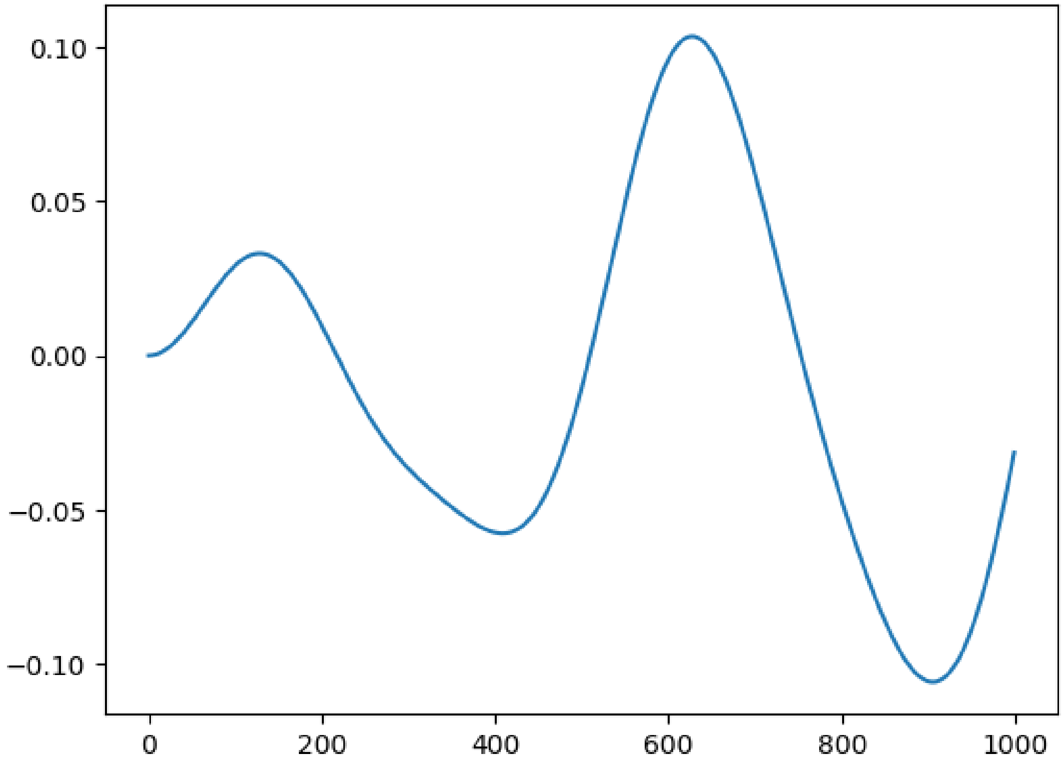
7. **Integrasi Numerik**
   Dengan memanfaatkan fungsi numerik dari percepatan sudut, sistem persamaan diferensial diselesaikan secara numerik menggunakan metode Runge-Kutta melalui fungsi odeint dari pustaka **SciPy**. Pendekatan ini menghasilkan evolusi waktu dari sudut dan kecepatan sudut masing-masing pendulum.

```
def dSdt(S, t):
  the1, z1, the2, z2 = S
  return [
      dthe1dt_f(z1),
      dz1dt_f(t, m, g, w, L1, L2, the1, the2, z1, z2),
      dthe2dt_f(z2),
      dz2dt_f(t, m, g, w, L1, L2, the1, the2, z1, z2),
  ]
```

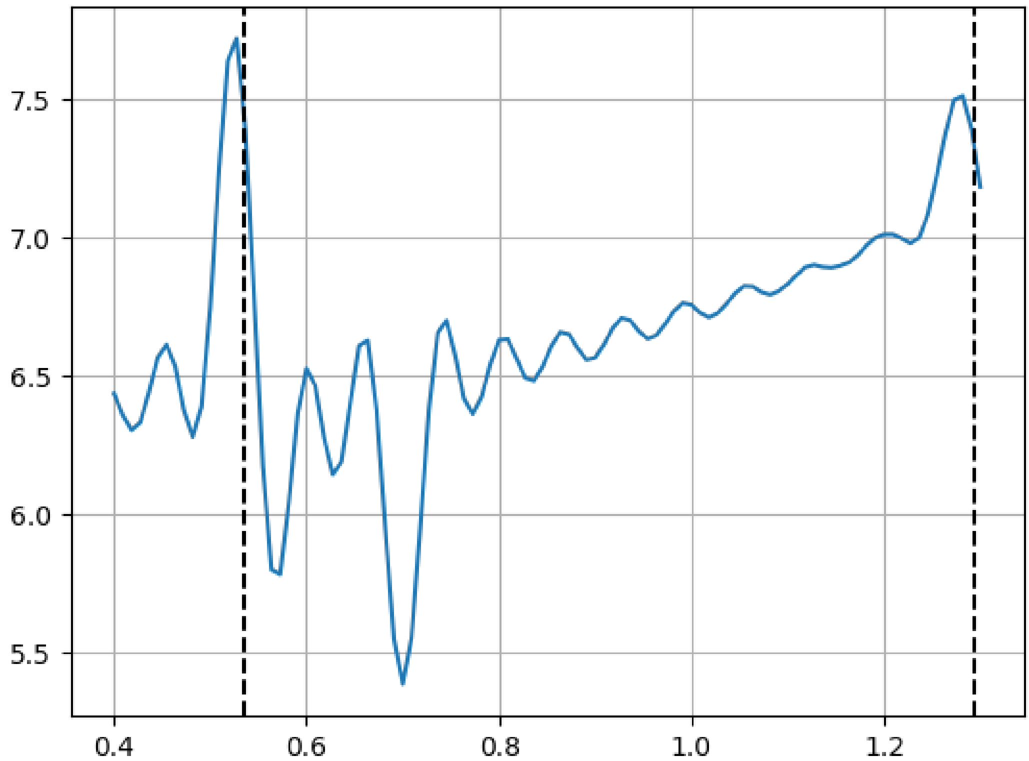
8. **Plot dan Visualisasi**
   Hasil integrasi numerik divisualisasikan. Grafik sudut terhadap waktu dibuat untuk memahami perilaku sistem, sedangkan animasi memperlihatkan gerak pendulum dalam ruang 2D. Animasi dibuat menggunakan **Matplotlib** dengan fungsi FuncAnimation.

```
ani = animation.FuncAnimation(fig, animate, frames=2000, interval=50)
HTML(ani.to_html5_video())
```

- **Analisis grafik dan animasi double pendulum**

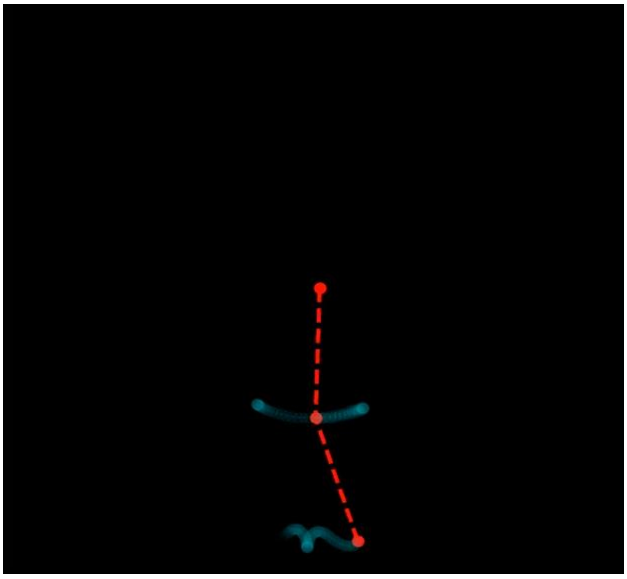


Pada grafik yang pertama ini menunjukkan osilasi sudut pada sistem pendulum double sebagai fungsi waktu, dengan pola gelombang yang mencerminkan sifat dinamis non-linear dari sistem. Perubahan amplitudo dan frekuensi osilasi mengindikasikan adanya interaksi kompleks antara dua pendulum dalam sistem, yang dipengaruhi oleh parameter seperti panjang pendulum L1 dan L2, gravitasi, dan kondisi awal sudut serta kecepatannya. Pola ini juga mencerminkan transfer energi antara komponen sistem, dengan kemungkinan resonansi terjadi pada frekuensi tertentu yang meningkatkan amplitudo osilasi. Analisis ini menunjukkan perilaku khas dari sistem mekanika klasik yang bergantung pada pendekatan Lagrangian.



Kemudian pada grafik ini menunjukkan hubungan antara energi kinetik rata-rata sistem terhadap frekuensi dari pendulum double. Pola grafik menggambarkan adanya variasi energi yang signifikan pada frekuensi tertentu, dengan dua garis vertikal putus-putus yang kemungkinan besar menunjukkan frekuensi resonansi.

Pada frekuensi ini, energi sistem mencapai nilai maksimum, menunjukkan bahwa sistem memasuki kondisi resonansi, di mana energi secara efisien ditransfer dalam osilasi. Di luar daerah resonansi, energi lebih stabil namun tetap menunjukkan pola fluktuasi kecil, mencerminkan sifat non-linear dari sistem dinamis. Grafik ini penting untuk memahami perilaku resonansi dalam sistem pendulum double dan pengaruhnya terhadap energi total sistem.



Dalam animasi double pendulum yang dihasilkan, dapat diamati bahwa pendulum pertama mempengaruhi pendulum kedua. Amplitudo dan lintasan gerakan setiap pendulum berubah seiring waktu akibat transfer energi antara keduanya. Gerakan ini dapat mencakup fase periodik pada awalnya, yang kemudian berkembang menjadi kaotik, di mana pola lintasan terlihat acak.

Jejak atau garis yang mengikuti pendulum menunjukkan dinamika jalur yang dilalui, dan pola tersebut mencerminkan interaksi energi kinetik dan potensial dalam sistem. Animasi ini memberikan gambaran visual yang kuat tentang bagaimana sistem mekanika klasik dapat menghasilkan perilaku kompleks dan kaotik, terutama dalam pengaturan multi-badan seperti double pendulum.