JAWABAN SOAL UAS

Nama : Nurmala Hayati Azzahra

Npm : 2410631330021

1. Sebuah pegas dengan konstanta pegas k = 60N / m diregangkan secara perlahan dari posisi keseimbangan. Gaya yang bekerja pada pegas akan mengikuti persamaan Hukum Hooke, yaitu F(x) = - k \* x di mana x adalah jarak regangan pegas dalam meter dari posisi keseimbangan.

a) Buat program Python untuk menghitung besar gaya F(x) pada pegas saat pegas diregangkan secara bertahap mulai dari posisi x = 0 meter hingga x = 0.5 meter dengan langkah Delta\*x = 0.02 meter.

b) Visualisasikan grafik gaya F(x) | terhadap posisi x menggunakan matplotlib.

INPUT

import numpy as np

import matplotlib.pyplot as plt

# Konstanta pegas

k = 60  # dalam N/m

# Fungsi gaya pegas menurut Hukum Hooke

def force(x):

    return -k \* x

# Rentang regangan pegas dan langkah

x\_values = np.arange(0, 0.52, 0.02)  # dari 0 hingga 0.5 meter dengan langkah 0.02 meter

# Hitung gaya untuk setiap x

f\_values = force(x\_values)

# Tampilkan hasil perhitungan

print("Posisi x (m)\tGaya F(x) (N)")

for x, f in zip(x\_values, f\_values):

    print(f"{x:.2f}\t\t{f:.2f}")

# Visualisasi grafik gaya terhadap posisi

plt.figure(figsize=(8, 6))

plt.plot(x\_values, f\_values, marker='o', linestyle='-', color='b', label='F(x) = -k\*x')

plt.title('Grafik Gaya Pegas terhadap Posisi Regangan')

plt.xlabel('Posisi Regangan x (m)')

plt.ylabel('Gaya F(x) (N)')

plt.grid(True)

plt.legend()

plt.show()

OUTPUT

Posisi x (m) Gaya F(x) (N)

0.00 -0.00

0.02 -1.20

0.04 -2.40

0.06 -3.60

0.08 -4.80

0.10 -6.00

0.12 -7.20

0.14 -8.40

0.16 -9.60

0.18 -10.80

0.20 -12.00

0.22 -13.20

0.24 -14.40

0.26 -15.60

0.28 -16.80

0.30 -18.00

0.32 -19.20

0.34 -20.40

0.36 -21.60

0.38 -22.80

0.40 -24.00

0.42 -25.20

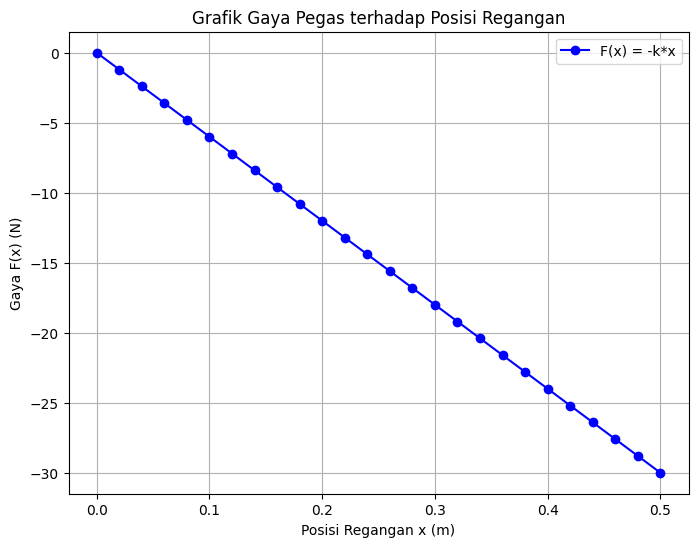
0.44 -26.40

0.46 -27.60

0.48 -28.80

0.50 -30.00

Grafik



1. Sebuah benda bermassa m digantung pada sebuah pegas dan melakukan gerak harmonis sederhana. Benda tersebut memiliki periode T = 10 detik dan amplitudo A = 5 cm. Tentukan persamaan posisi benda sebagai fungsi waktu dan buat program Python untuk menghitung posisi benda pada setiap waktu dari 0 hingga 50 detik. Kemudian visualisasikan gerak harmonis sederhana ini dalam bentuk grafik posisi terhadap waktu.

INPUT

import numpy as np

import matplotlib.pyplot as plt

# Parameter gerak harmonis sederhana

T = 10  # periode dalam detik

A = 0.05  # amplitudo dalam meter (5 cm)

# Frekuensi sudut (omega)

omega = 2 \* np.pi / T

# Fungsi posisi sebagai fungsi waktu

def position(t):

    return A \* np.cos(omega \* t)

# Waktu dari 0 hingga 50 detik dengan interval 0.1 detik

t\_values = np.arange(0, 50.1, 0.1)

# Hitung posisi untuk setiap waktu

x\_values = position(t\_values)

# Tampilkan hasil perhitungan

print("Waktu (s)\tPosisi x (m)")

for t, x in zip(t\_values, x\_values):

    print(f"{t:.1f}\t\t{x:.3f}")

# Visualisasi grafik posisi terhadap waktu

plt.figure(figsize=(10, 6))

plt.plot(t\_values, x\_values, label='x(t) = A\*cos(omega\*t)', color='g')

plt.title('Gerak Harmonis Sederhana')

plt.xlabel('Waktu (s)')

plt.ylabel('Posisi x (m)')

plt.grid(True)

plt.legend()

plt.show()

OUTPUT

Waktu (s) Posisi x (m)

0.0 0.050

0.1 0.050

0.2 0.050

0.3 0.049

0.4 0.048

0.5 0.048

0.6 0.046

0.7 0.045

0.8 0.044

0.9 0.042

1.0 0.040

1.1 0.039

1.2 0.036

1.3 0.034

1.4 0.032

1.5 0.029

1.6 0.027

1.7 0.024

1.8 0.021

1.9 0.018

2.0 0.015

2.1 0.012

2.2 0.009

2.3 0.006

2.4 0.003

2.5 0.000

2.6 -0.003

2.7 -0.006

2.8 -0.009

2.9 -0.012

3.0 -0.015

3.1 -0.018

3.2 -0.021

3.3 -0.024

3.4 -0.027

3.5 -0.029

3.6 -0.032

3.7 -0.034

3.8 -0.036

3.9 -0.039

4.0 -0.040

4.1 -0.042

4.2 -0.044

4.3 -0.045

4.4 -0.046

4.5 -0.048

4.6 -0.048

4.7 -0.049

4.8 -0.050

4.9 -0.050

5.0 -0.050

5.1 -0.050

5.2 -0.050

5.3 -0.049

5.4 -0.048

5.5 -0.048

5.6 -0.046

5.7 -0.045

5.8 -0.044

5.9 -0.042

6.0 -0.040

6.1 -0.039

6.2 -0.036

6.3 -0.034

6.4 -0.032

6.5 -0.029

6.6 -0.027

6.7 -0.024

6.8 -0.021

6.9 -0.018

7.0 -0.015

7.1 -0.012

7.2 -0.009

7.3 -0.006

7.4 -0.003

7.5 -0.000

7.6 0.003

7.7 0.006

7.8 0.009

7.9 0.012

8.0 0.015

8.1 0.018

8.2 0.021

8.3 0.024

8.4 0.027

8.5 0.029

8.6 0.032

8.7 0.034

8.8 0.036

8.9 0.039

9.0 0.040

9.1 0.042

9.2 0.044

9.3 0.045

9.4 0.046

9.5 0.048

9.6 0.048

9.7 0.049

9.8 0.050

9.9 0.050

10.0 0.050

10.1 0.050

10.2 0.050

10.3 0.049

10.4 0.048

10.5 0.048

10.6 0.046

10.7 0.045

10.8 0.044

10.9 0.042

11.0 0.040

11.1 0.039

11.2 0.036

11.3 0.034

11.4 0.032

11.5 0.029

11.6 0.027

11.7 0.024

11.8 0.021

11.9 0.018

12.0 0.015

12.1 0.012

12.2 0.009

12.3 0.006

12.4 0.003

12.5 0.000

12.6 -0.003

12.7 -0.006

12.8 -0.009

12.9 -0.012

13.0 -0.015

13.1 -0.018

13.2 -0.021

13.3 -0.024

13.4 -0.027

13.5 -0.029

13.6 -0.032

13.7 -0.034

13.8 -0.036

13.9 -0.039

14.0 -0.040

14.1 -0.042

14.2 -0.044

14.3 -0.045

14.4 -0.046

14.5 -0.048

14.6 -0.048

14.7 -0.049

14.8 -0.050

14.9 -0.050

15.0 -0.050

15.1 -0.050

15.2 -0.050

15.3 -0.049

15.4 -0.048

15.5 -0.048

15.6 -0.046

15.7 -0.045

15.8 -0.044

15.9 -0.042

16.0 -0.040

16.1 -0.039

16.2 -0.036

16.3 -0.034

16.4 -0.032

16.5 -0.029

16.6 -0.027

16.7 -0.024

16.8 -0.021

16.9 -0.018

17.0 -0.015

17.1 -0.012

17.2 -0.009

17.3 -0.006

17.4 -0.003

17.5 -0.000

17.6 0.003

17.7 0.006

17.8 0.009

17.9 0.012

18.0 0.015

18.1 0.018

18.2 0.021

18.3 0.024

18.4 0.027

18.5 0.029

18.6 0.032

18.7 0.034

18.8 0.036

18.9 0.039

19.0 0.040

19.1 0.042

19.2 0.044

19.3 0.045

19.4 0.046

19.5 0.048

19.6 0.048

19.7 0.049

19.8 0.050

19.9 0.050

20.0 0.050

20.1 0.050

20.2 0.050

20.3 0.049

20.4 0.048

20.5 0.048

20.6 0.046

20.7 0.045

20.8 0.044

20.9 0.042

21.0 0.040

21.1 0.039

21.2 0.036

21.3 0.034

21.4 0.032

21.5 0.029

21.6 0.027

21.7 0.024

21.8 0.021

21.9 0.018

22.0 0.015

22.1 0.012

22.2 0.009

22.3 0.006

22.4 0.003

22.5 0.000

22.6 -0.003

22.7 -0.006

22.8 -0.009

22.9 -0.012

23.0 -0.015

23.1 -0.018

23.2 -0.021

23.3 -0.024

23.4 -0.027

23.5 -0.029

23.6 -0.032

23.7 -0.034

23.8 -0.036

23.9 -0.039

24.0 -0.040

24.1 -0.042

24.2 -0.044

24.3 -0.045

24.4 -0.046

24.5 -0.048

24.6 -0.048

24.7 -0.049

24.8 -0.050

24.9 -0.050

25.0 -0.050

25.1 -0.050

25.2 -0.050

25.3 -0.049

25.4 -0.048

25.5 -0.048

25.6 -0.046

25.7 -0.045

25.8 -0.044

25.9 -0.042

26.0 -0.040

26.1 -0.039

26.2 -0.036

26.3 -0.034

26.4 -0.032

26.5 -0.029

26.6 -0.027

26.7 -0.024

26.8 -0.021

26.9 -0.018

27.0 -0.015

27.1 -0.012

27.2 -0.009

27.3 -0.006

27.4 -0.003

27.5 -0.000

27.6 0.003

27.7 0.006

27.8 0.009

27.9 0.012

28.0 0.015

28.1 0.018

28.2 0.021

28.3 0.024

28.4 0.027

28.5 0.029

28.6 0.032

28.7 0.034

28.8 0.036

28.9 0.039

29.0 0.040

29.1 0.042

29.2 0.044

29.3 0.045

29.4 0.046

29.5 0.048

29.6 0.048

29.7 0.049

29.8 0.050

29.9 0.050

30.0 0.050

30.1 0.050

30.2 0.050

30.3 0.049

30.4 0.048

30.5 0.048

30.6 0.046

30.7 0.045

30.8 0.044

30.9 0.042

31.0 0.040

31.1 0.039

31.2 0.036

31.3 0.034

31.4 0.032

31.5 0.029

31.6 0.027

31.7 0.024

31.8 0.021

31.9 0.018

32.0 0.015

32.1 0.012

32.2 0.009

32.3 0.006

32.4 0.003

32.5 -0.000

32.6 -0.003

32.7 -0.006

32.8 -0.009

32.9 -0.012

33.0 -0.015

33.1 -0.018

33.2 -0.021

33.3 -0.024

33.4 -0.027

33.5 -0.029

33.6 -0.032

33.7 -0.034

33.8 -0.036

33.9 -0.039

34.0 -0.040

34.1 -0.042

34.2 -0.044

34.3 -0.045

34.4 -0.046

34.5 -0.048

34.6 -0.048

34.7 -0.049

34.8 -0.050

34.9 -0.050

35.0 -0.050

35.1 -0.050

35.2 -0.050

35.3 -0.049

35.4 -0.048

35.5 -0.048

35.6 -0.046

35.7 -0.045

35.8 -0.044

35.9 -0.042

36.0 -0.040

36.1 -0.039

36.2 -0.036

36.3 -0.034

36.4 -0.032

36.5 -0.029

36.6 -0.027

36.7 -0.024

36.8 -0.021

36.9 -0.018

37.0 -0.015

37.1 -0.012

37.2 -0.009

37.3 -0.006

37.4 -0.003

37.5 -0.000

37.6 0.003

37.7 0.006

37.8 0.009

37.9 0.012

38.0 0.015

38.1 0.018

38.2 0.021

38.3 0.024

38.4 0.027

38.5 0.029

38.6 0.032

38.7 0.034

38.8 0.036

38.9 0.039

39.0 0.040

39.1 0.042

39.2 0.044

39.3 0.045

39.4 0.046

39.5 0.048

39.6 0.048

39.7 0.049

39.8 0.050

39.9 0.050

40.0 0.050

40.1 0.050

40.2 0.050

40.3 0.049

40.4 0.048

40.5 0.048

40.6 0.046

40.7 0.045

40.8 0.044

40.9 0.042

41.0 0.040

41.1 0.039

41.2 0.036

41.3 0.034

41.4 0.032

41.5 0.029

41.6 0.027

41.7 0.024

41.8 0.021

41.9 0.018

42.0 0.015

42.1 0.012

42.2 0.009

42.3 0.006

42.4 0.003

42.5 -0.000

42.6 -0.003

42.7 -0.006

42.8 -0.009

42.9 -0.012

43.0 -0.015

43.1 -0.018

43.2 -0.021

43.3 -0.024

43.4 -0.027

43.5 -0.029

43.6 -0.032

43.7 -0.034

43.8 -0.036

43.9 -0.039

44.0 -0.040

44.1 -0.042

44.2 -0.044

44.3 -0.045

44.4 -0.046

44.5 -0.048

44.6 -0.048

44.7 -0.049

44.8 -0.050

44.9 -0.050

45.0 -0.050

45.1 -0.050

45.2 -0.050

45.3 -0.049

45.4 -0.048

45.5 -0.048

45.6 -0.046

45.7 -0.045

45.8 -0.044

45.9 -0.042

46.0 -0.040

46.1 -0.039

46.2 -0.036

46.3 -0.034

46.4 -0.032

46.5 -0.029

46.6 -0.027

46.7 -0.024

46.8 -0.021

46.9 -0.018

47.0 -0.015

47.1 -0.012

47.2 -0.009

47.3 -0.006

47.4 -0.003

47.5 -0.000

47.6 0.003

47.7 0.006

47.8 0.009

47.9 0.012

48.0 0.015

48.1 0.018

48.2 0.021

48.3 0.024

48.4 0.027

48.5 0.029

48.6 0.032

48.7 0.034

48.8 0.036

48.9 0.039

49.0 0.040

49.1 0.042

49.2 0.044

49.3 0.045

49.4 0.046

49.5 0.048

49.6 0.048

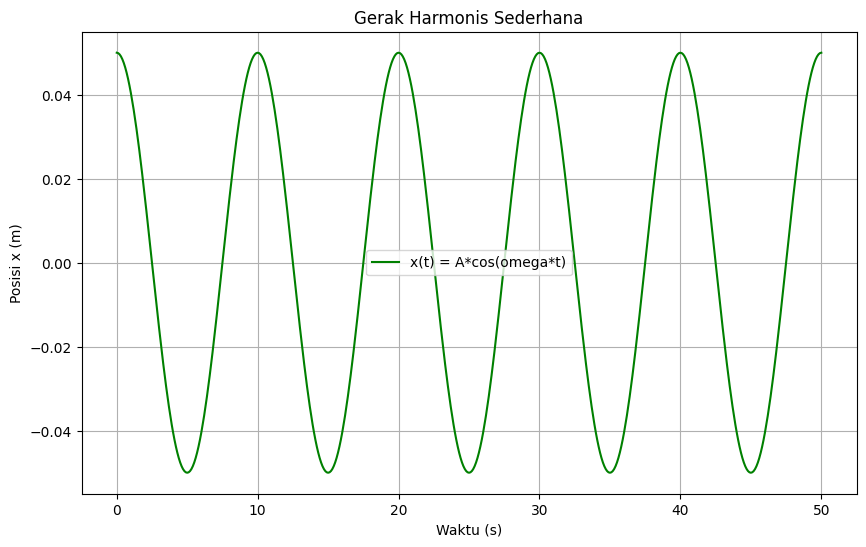
49.7 0.049

49.8 0.050

49.9 0.050

50.0 0.050

GRAFIK



1. Sebuah benda dilempar vertikal ke atas dengan kecepatan awal 50m / s buatlah program python dan grafiknya untuk:

a) Berakhiran NPM ganjil, gambaran grafik gerak bendanya sampai tinggi maksimum

b) Berakhiran NPM genap, Gambaran grafik gerak bendanya sampai kembali lagi ke posisi awal

INPUT

import numpy as np

import matplotlib.pyplot as plt

# Parameter gerak

v0 = 50  # kecepatan awal dalam m/s

g = 9.8  # percepatan gravitasi dalam m/s^2

# Fungsi posisi sebagai fungsi waktu untuk benda yang dilempar ke atas

def position(t):

    return v0 \* t - 0.5 \* g \* t\*\*2

# Kondisi untuk tinggi maksimum

# Waktu untuk mencapai tinggi maksimum: v = 0, t = v0/g

t\_max = v0 / g

# Kondisi untuk kembali ke posisi awal

# Waktu total: posisi y = 0, 0 = v0 \* t - 0.5 \* g \* t^2

# t = 2 \* t\_max

t\_total = 2 \* t\_max

# Rentang waktu untuk grafik

# Ganti nilai akhir sesuai dengan tipe NPM (ganjil/genap)

# Jika NPM ganjil, gunakan t\_max, jika genap gunakan t\_total

npm\_genap = True  # Ubah ke False jika NPM ganjil

if npm\_genap:

    t\_values = np.linspace(0, t\_total, 500)

else:

    t\_values = np.linspace(0, t\_max, 500)

# Hitung posisi untuk setiap waktu

y\_values = position(t\_values)

# Tampilkan grafik

plt.figure(figsize=(10, 6))

plt.plot(t\_values, y\_values, label='y(t) = v0\*t - 0.5\*g\*t^2', color='b')

plt.axhline(0, color='black', linewidth=0.8, linestyle='--')  # garis tanah

plt.title('Grafik Gerak Vertikal Benda')

plt.xlabel('Waktu (s)')

plt.ylabel('Tinggi (m)')

plt.grid(True)

plt.legend()

plt.show()

1. Sebuah gaya tak konstan memenuhi persamaan F = (2x ^ 2 + 4x + 2) N bekerja pada benda sehingga benda tersebut berpindah sejauh 30 meter. Buatlah program python dari persoalan tersebut dan visualisasikan dalam bentuk grafik.

INPUT

import numpy as np

import matplotlib.pyplot as plt

# Fungsi gaya sebagai fungsi perpindahan

# F(x) = 2x^2 + 4x + 2

def force(x):

    return 2 \* x\*\*2 + 4 \* x + 2

# Rentang perpindahan

x\_values = np.linspace(0, 30, 500)  # dari 0 hingga 30 meter dengan 500 titik data

# Hitung gaya untuk setiap perpindahan

f\_values = force(x\_values)

# Tampilkan grafik

plt.figure(figsize=(10, 6))

plt.plot(x\_values, f\_values, label='F(x) = 2x^2 + 4x + 2', color='r')

plt.title('Grafik Gaya terhadap Perpindahan')

plt.xlabel('Perpindahan (m)')

plt.ylabel('Gaya (N)')

plt.grid(True)

plt.legend()

plt.show()