

Praktikum Fisika Komputasi

Nama : Attala Muflih Gumilang

Nim : 1227030007

Jurusan : Fisika

Analisis Kode Program Yang Di Gunakan Untuk Menghasilkan Hasil Integral

Kode di bawah digunakan untuk menghitung integral dari fungsi $f(x)=x^2\cos(x)+3\sin(2x)$ pada rentang $[0,\pi]$ menggunakan fungsi `quad()` dari pustaka `scipy`, serta menampilkan grafik fungsi tersebut dengan `matplotlib`.

Pertama-tama, ditetapkan nilai awal dan akhir interval, yaitu $x_{\text{start}} = 0$ dan $x_{\text{stop}} = \pi$, dengan jarak antar titik sebesar $x_{\text{steps_interval}} = 0.01$. Kemudian, array `x_values` dibuat menggunakan `np.arange(x_start, x_stop, x_steps_interval)`, yang menghasilkan nilai-nilai x pada interval tersebut.

Fungsi $f(x)$ didefinisikan sebagai `y_values = x_values**2 * np.cos(x_values) + 3 * np.sin(2 * x_values)`, untuk mendapatkan nilai y di setiap titik x dalam array `x_values`. Untuk melakukan integrasi, fungsi `lambda integration_function = lambda x: x**2 * np.cos(x) + 3 * np.sin(2 * x)` didefinisikan sesuai dengan bentuk fungsi $f(x)$ yang diberikan.

Integrasi dilakukan menggunakan `integrate.quad(integration_function, x_start, x_stop)`, dan hasilnya disimpan dalam variabel `integral`. Kemudian, nilai integral ditampilkan dengan `print("Integral Value:")` dan `print(integral)`, sehingga bisa melihat hasilnya. Grafik dari fungsi $f(x)$ kemudian digambar menggunakan `plotlib.plot(x_values, y_values)`, dengan sumbu diberi label serta ditambahkan judul. Fungsi `plotlib.show()` digunakan untuk menampilkan grafik fungsi pada rentang $[0,\pi]$.

```

import numpy as np
from scipy import integrate
import matplotlib.pyplot as plt

# Define parameters
x_start = 0 # Start of the interval
x_stop = np.pi # End of the interval
x_steps_interval = 0.01 # Step size

# Define an array of data points
x_values = np.arange(x_start, x_stop, x_steps_interval)

# Define the function f(x) = x^2 * cos(x) + 3 * sin(2x)
y_values = x_values**2 * np.cos(x_values) + 3 * np.sin(2 * x_values)

# Plot the function curve
plt.plot(x_values, y_values)

# Define a lambda function for integration
integration_function = lambda x: x**2 * np.cos(x) + 3 * np.sin(2 * x)

# Calculate the integral (ignoring error)
integral, _ = integrate.quad(integration_function, x_start, x_stop)

# Print the integration result
print("Integral Value:")
print(integral)

# Display the plot
plt.xlabel('x')
plt.ylabel('f(x)')
plt.title('Plot of f(x) = x^2 * cos(x) + 3 * sin(2x)')
plt.show()

```

Figure 1



(x, y) = (2.266, 3.51)

