

Praktikum Fisika Komputasi

Nama :Attala Muflih Gumilang

Nim :1227030007

Jurusan :Fisika

Penjelasan program

1. Trapzoid

Pada kodingan ini, untuk numpy di gunakan untuk komputasi numerik, yaitu untuk membuat array dan operasi matematika. Sedangkan matplotlib.pyplot di gunakan untuk membuat visualisasi grafik.

Selanjutnya ada definisi atau persamaan yang di gunakan, lalu di bawahnya terdapat sebuah nilai yang di ketahui, lalu kodingan selanjutnya untuk metode trapezoid adalah kodingan untuk menghitung integralnya.

```
✓ [1] # Mengimpor Library
1d import numpy as np
import matplotlib.pyplot as plt
```

```
✓ #Integral
0d def func(x):
    return x**-3 + np.cos(x)
a = 1.0
b = 5.0
```

```
✓ # Metode Trapezoid
0d n = 1000 # Jumlah grid
dx = (b - a) / (n - 1)
x = np.linspace(a, b, n)

sigma = 0
for i in range(1, n - 1):
    sigma += func(x[i])

hasil = 0.5 * dx * (func(x[0]) + 2 * sigma + func(x[-1]))

print(hasil)
```

```
🔄 -1.3203888525573348
```

```
✓ [4] xp = np.linspace(a, b, 1000)
0d plt.plot(xp, func(xp))
plt.show()
```

Kodingan di bawah ini adalah kodingan untuk menampilkan sebuah grafik yang berbeda beda,ada visualisasi dengan batang,visualisasi fungsi dengan area di arsir,dan plot fungsi.

```
1 1 ▶ xp =np.linspace(a,b,1000)
    plt.plot(xp,func(xp))
    plt.show()
```

```
✓ 9 d [5] xp = np.linspace(a,b,1000)
        plt.plot(xp,func(xp))

        for i in range (n):
            plt.bar(x[i],func(x[i]), align= 'edge',width= 0.000001, edgecolor='red')

        plt.show()
```

```
✓ 9 d ▶ xp =np.linspace(a,b,1000)
        plt.plot(xp,func(xp))

        for i in range (n):
            plt.bar(x[i],func(x[i]),align='edge',width=0.000001, edgecolor='red')


        plt.fill_between(x,func(x),color='yellow',alpha=0.5)

        plt.show()
```

2. Simpson 1/3

Kodingan yang pertama yaitu memasukan library, ada numpy digunakan untuk komputasi numerik, seperti membuat array dan melakukan operasi matematika. Dan matplotlib.pyplot digunakan untuk membuat visualisasi grafik. Di kodingan selanjutnya terdapat sebuah nilai fungsi yang di ketahui dan pendefinisian dari batas intergral dan jumlah titik.

Selanjutnya ada simson's rule,untuk mengkoreksi jumlah titik dengan metode Simpson dan membuat grid dan jarak antar titik.Lalu di bawahnya kodingan untuk menghitung integral menggunakan metode Simpson.Selanjutnya juga terdapat ada visualisasi grafik dan bar chart,kodingan tersebut untuk membuat grafik dan menampilkan sebuah grafik yang telah di buat.

```
id  #Mengimport Library
import numpy as np
import matplotlib.pyplot as plt

#Integral
def func(x):
    return x**-3 + np.cos(x)
a = 1.0
b = 5.0
n = 1000

#Simpson's Rule
if n % 2 == 0:
    n+=1
x= np.linspace(a,b,n)
dx=(x[-1]-x[0])/(n-1)

#Menghitung Integral Menggunakan Metode Simpson
hasil = func(x[0])+func(x[-1]))

for i in range (1,n-1,2):
    hasil +=4*func(x[i])

for i in range(2,n-2,2):
    hasil +=2*func(x[i])

hasil*=dx/3

#visualisasi grafik dan bar
xp=np.linspace(a,b,1000)
plt.plot(xp,func(xp))

for i in range(n):
    plt.bar(x[i],func(x[i]),align='edge',width=dx,color='red',edgecolor='black')
plt.show()
print(hasil)
```

3. Metode yang lebih efektif

Metode yang lebih efektif yaitu menurut saya adalah metode Simpson, karena di dalam kodingan. keakuratan lebih tinggi dengan jumlah grid yang lumayan sedikit. Dan juga dengan persamaan dan Simpson rule yang membuat bahwa metode ini lebih efektif