**Nama : Aszka Sazkia Juliana Sipa**

**Nim : 1227030006**

## PENYELESAIAN METODE EKSAK

Nama : Aszka Sazkia J.S

Nim : 1227030006

$$\int_1^5 x^{-3} + \cos(x)\,dx$$

Solusi

$$\int_1^5 x^{-3} + \cos(x)\,dx = \int_1^5 x^{-3}\,dx + \int_1^5 \cos(x)\,dx$$

$$\int_1^5 x^{-3} + \cos(x)\,dx = \left[-\frac{1}{2}x^{-2}\right|_1^5 + \left[\sin(x)\right|_1^5$$

$$= \left\{\left(-\frac{1}{50}\right) - \left(\frac{1}{2}\right)\right\} + \left\{(\sin(5)) - \sin(1))\right\}$$

$$= -0{,}02 - 0{,}5 + (-0{,}97892427) - 0{,}8414709$$

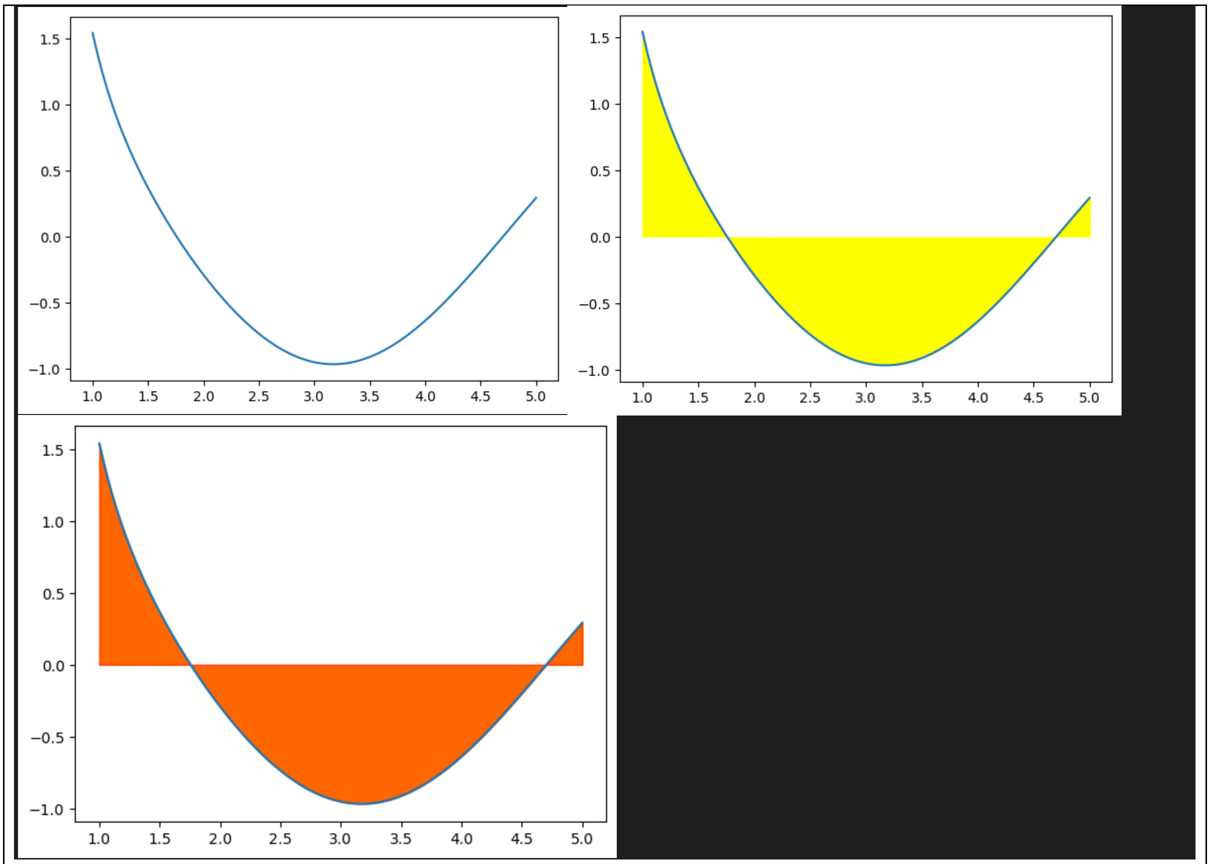
$$= -1{,}320395 \;//$$

## PENYELESAIAN METODE TRAPEZOID

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

#Integral
def func(x):                  #Nama fungsi
    return (x**(-3))+np.cos(x) #Fungsi yang akan diintegralkan
a = 1.0                       #Batas bawah
b = 5.0                       #Batas atas

#Metode Trapezoid
```

```
n = 500
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[i])+2*sigma+func(x[-1]))

print(hasil)

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

#Grafik 2
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='yellow')

plt.show()

#Grafik 3
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='yellow')

plt.fill_between(x,func(x),color='red', alpha=0.6)

plt.show()
```

```
Hasil perhitungan :  -1.325404840477387
```

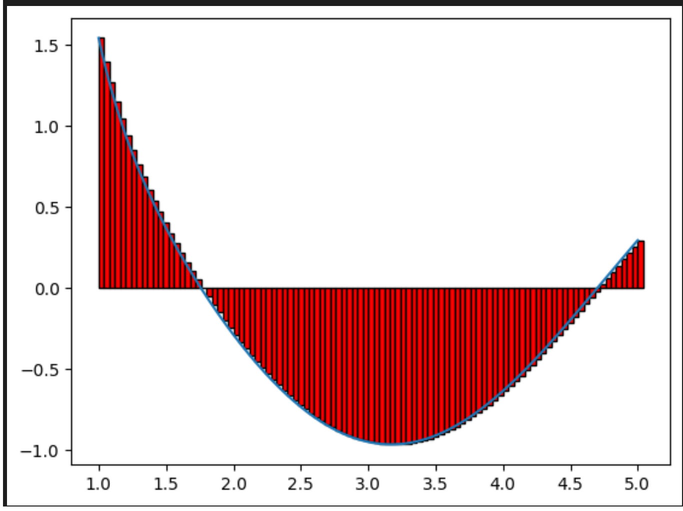



**PENYELESAIAN METODE SIMPSON 1/3**

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

#Fungsi Integral
def func(x):                  #Nama fungsi
    return (x**(-3))+np.cos(x)  #Fungsi yang akan diintegralkan
a = 1.0                       #Batas bawah
b = 5.0                       #Batas atas
n = 100                       #Jumlah grid, harus ganjil untuk metode simpson
#Simpson's Rule
if n % 2 == 0:  #Jika n nya dibagi 2 dan hasilnya genap, maka tidak memenuhi aturan Simpson
    n += 1      #Jika n genap, tambah 1 agar menjadi ganjil

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])    #Tambah f(a) dan f(b)

for i in range(1,n-1,2):
```

```
    hasil += 4*func(x[i])        #Untuk indeks ganjil

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

hasil*= dx/3                     #Faktor 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)
```

Hasil perhitungan: -1.3203944385483368




**ANALISIS DAN PEMBAHASAN**

Berdasarkan hasil dari perhitungan secara eksak didapatkan hasil perhitungan penyelesaian integral sebesar -1,320395 ,selanjutnya menggunakan metode trapezoid sebesar -1, 325404840477387 dan menggunakan metode simpson didapatkan hasil sebesar -1,320394 Berdasarkan hasil perhitungan yang didapatkan, dengan menggunakan metode simpson didapatkan hasil yang hampir sama dengan hasil dari perhitungan eksak. Pada metode trapezoid digunakan n(jumlah grid) sebanyak 500 tetapi hasilnya belum terlalu akurat, sementara pada metode simpson hanya digunakan n sebanyak 100 tapi hasilnya hampir sama. Sehingga metode simpson dirasa lebih efektif dan akurat untuk menyelesaikan persamaan integral secara numerik.