Nama: Ririn Anastasya

NIM: 1227030030

Modul 4 Integral Metode Numerik

- Metode Eksak

Nama : Ririn Anastasya
NIM : 1227030030

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

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

• $\int_1^5 x^{-3}\ dx$

$$\int x^{-3}\, dx = \frac{x^{-2}}{-2} = -\frac{1}{2x^2}$$

$$\left[-\frac{1}{2x^2}\right]_1^5 = \left(-\frac{1}{2(5)^2}\right) - \left(-\frac{1}{2(1)^2}\right)$$

$$-\frac{1}{2(5)^2} = -\frac{1}{2\times 25} = -\frac{1}{50}$$

$$-\frac{1}{2(1)^2} = -\frac{1}{2\times 1} = -\frac{1}{2}$$

$$= -\frac{1}{50} + \frac{1}{2} = \frac{1}{2} - \frac{1}{50} = \frac{25}{50} - \frac{1}{50} = \frac{24}{50} = \frac{12}{25} = 0{,}48$$

• $\int_1^5 \cos(x)\ dx$

$$\int \cos(x)\, dx = \sin(x)$$

$$[\sin(x)]_1^5 = \sin(5) - \sin(1)$$

$\sin(5) \approx -0{,}9589$

$\sin(1) \approx 0{,}8415$

$\sin(5) - \sin(1) \approx -0{,}9589 - 0{,}8415 = -1{,}8004$

$$: \int_1^5 x^{-3} + \cos(x)\ dx = 0{,}48 - 1{,}8004 = -1{,}3204$$

Melalui metode eksak hasil sebenarnya dari integral dapat dihitung secara analitis atau dengan metode numerik yang lebih akurat. Hasil yang diperoleh adalah -1.3204.

- Metode Trapezoid

[2]

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

[38]

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

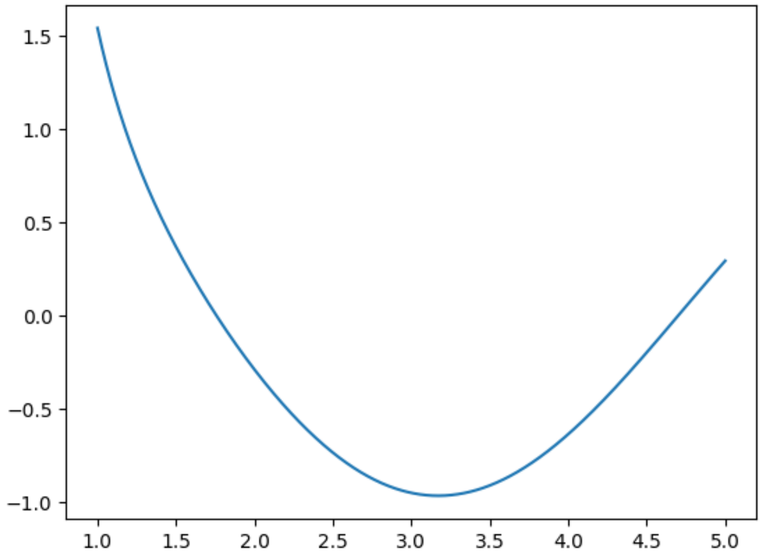
Daftar isi

```
#Metode Trapezoid
n = 10
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)
```

```
-1.24412040603867
```

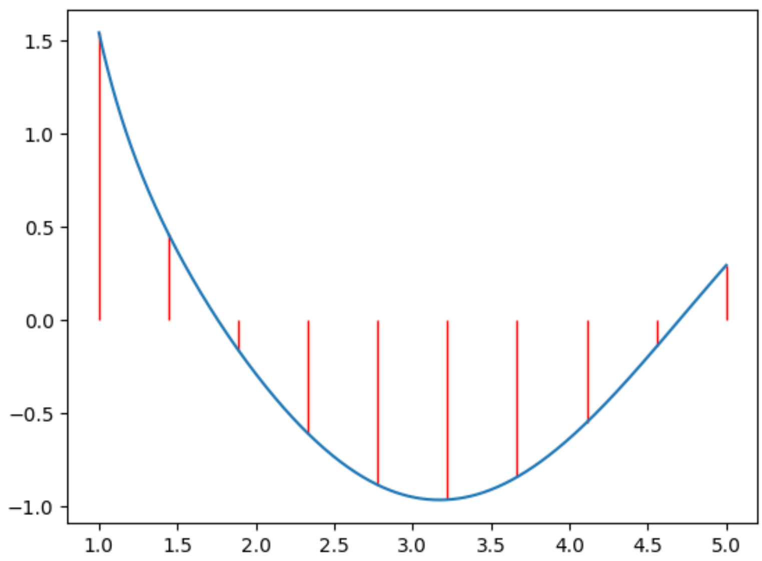
[36]

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

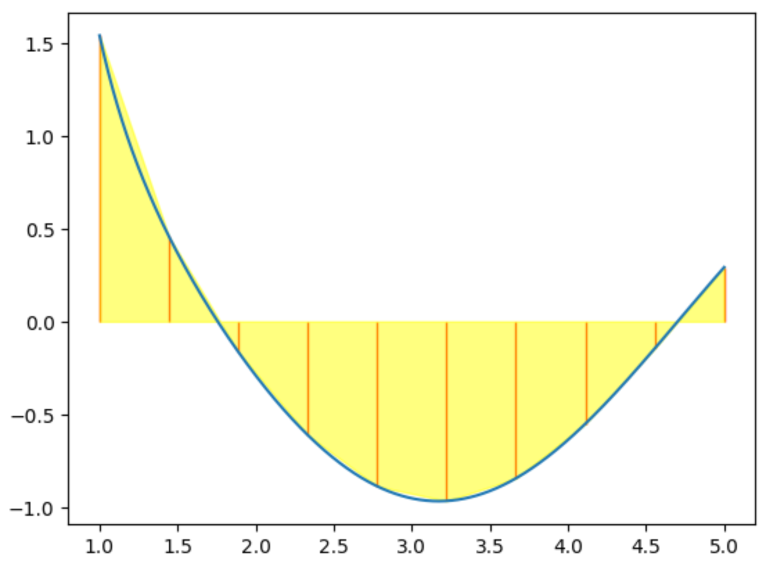


[37]

```
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()
```




```python
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()
```

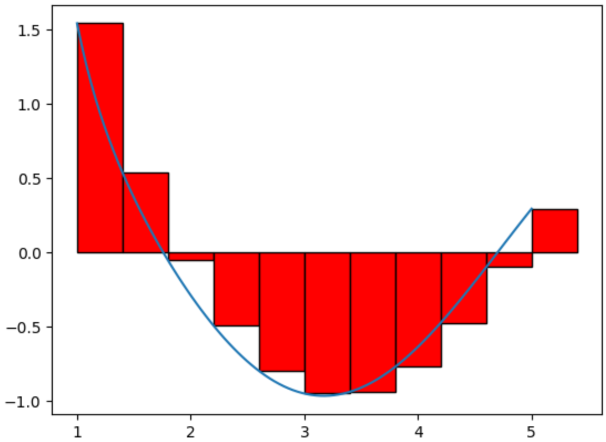


Di metode ini, luas di bawah kurva dihitung dengan membagi area menjadi trapezoid atau trapesium. Hasil yang diperoleh adalah sekitar -1.24412.

- Metode Simpson 1/3

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

# Fungsi yang akan diintegralkan
def func(x):
    return (x**-3)+np.cos(x)

# Batas integrasi
a = 1.0  # Batas bawah
b = 5.0  # Batas atas
n = 10  # Jumlah grid, harus ganjil untuk metode Simpson
```

```python
# Simpson's Rule
if n % 2 == 0:
    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, 100)
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)
```




```
-1.3155721992540337
```

Metode ini menghitung luas di bawah kurva dengan memecah area menjadi bagian-bagian kecil berbentuk parabola. Hasil yang diperoleh dari metode ini sekitar -1.31557.

Metode Simpson 1/3 dan metode Trapezoid adalah dua cara berbeda untuk menghitung luas di bawah kurva atau dalam menghitung nilai integral dari suatu fungsi. Hasil perhitungan menggunakan metode Simpson 1/3 mendekati -1.31557, sedangkan metode Trapezoid adalah sekitar -1.24412. Ketika dibandingkan dengan hasil yang didapat dari metode eksak, yaitu -1.3204, terlihat bahwa metode Simpson 1/3 lebih akurat. Ini terjadi karena metode Simpson 1/3 memperhitungkan bentuk lengkung dari kurva dengan menggunakan parabola sehingga lebih cocok untuk fungsi yang tidak lurus.

Sedangkan, metode Trapezoid menggunakan garis lurus yang menghubungkan titik-titik di bawah kurva. Akibatnya, hasilnya kurang akurat untuk fungsi yang berkelokkelok atau berbentuk kompleks. Untuk hasil yang lebih presisi, metode Simpson 1/3 lebih tepat karena mampu menangkap bentuk asli dari kurva dengan lebih baik.