JAWABAN

PRAKTIKUM FISIKA KOMPUTASI

INTEGRAL METODE NUMERIK

Nama : Gilang Pratama Putra Siswanto

NIM : 1227030017

Prodi : Fisika (Angkatan 2022)

Kode Tugas : Tugas 4 – Integral Metode Numerik

BABAK 1 – Penyelesaian Soal Integral Melalui 3 Metode

Disajikan soal integral sebagai berikut.

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

Dalam menyelesaiakan soal tersebut, digunakan 3 metode dalam mencari solusi dari integral tersebut yaitu:

a) Metode eksak
b) Metode trapezoid
c) Metode Simpson 1/3

a) Metode Eksak

Penyelesaian soal integral di atas telah berhasil diselesaikan secara eksak dengan tahapan penyelesaian sebagai berikut.

No.

Date

PRAKTIKUM FISIKA KOMPUTASI - TUGAS 4

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

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

$$= \int_1^5 \frac{1}{-3+1} x^{-3+1}\, dx + \int_1^5 \cos(x)\, dx$$

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

$$= -\frac{1}{2}(5)^{-2} + \frac{1}{2}(1)^{-2} + \sin(5) - \sin(1)$$

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

$$= \frac{-1}{50} + \frac{25}{50} - 0{,}958 - 0{,}841$$

$$= \frac{24}{50} - 0{,}9589 - 0{,}8414$$

$$= -1{,}3203$$

Berdasarkan perhitungan secara eksak (manual), maka diperoleh nilai akhir yaitu di -1,3203.

b) Metode Trapezoid

Disusun algoritma pemrograman metode integrasi trapezoid untuk mencari nilai hasil integral dari fungsi yang diberikan, kemudian melakukan *plotting* untuk menghasilkan visualisasi dari tahapan integrasi yang diberikan menggunakan metode trapezoid. Adapun algoritma pemrograman serta hasil yang diperoleh ditampilkan sebagai berikut.

```
# Nama    : Gilang Pratama Putra Siswanto
# NIM     : 1227030017
# Tugas 4 Inteagral Metode Numerik - Praktikum Fisika
Komputasi

import numpy as np
import matplotlib.pyplot as plt
```

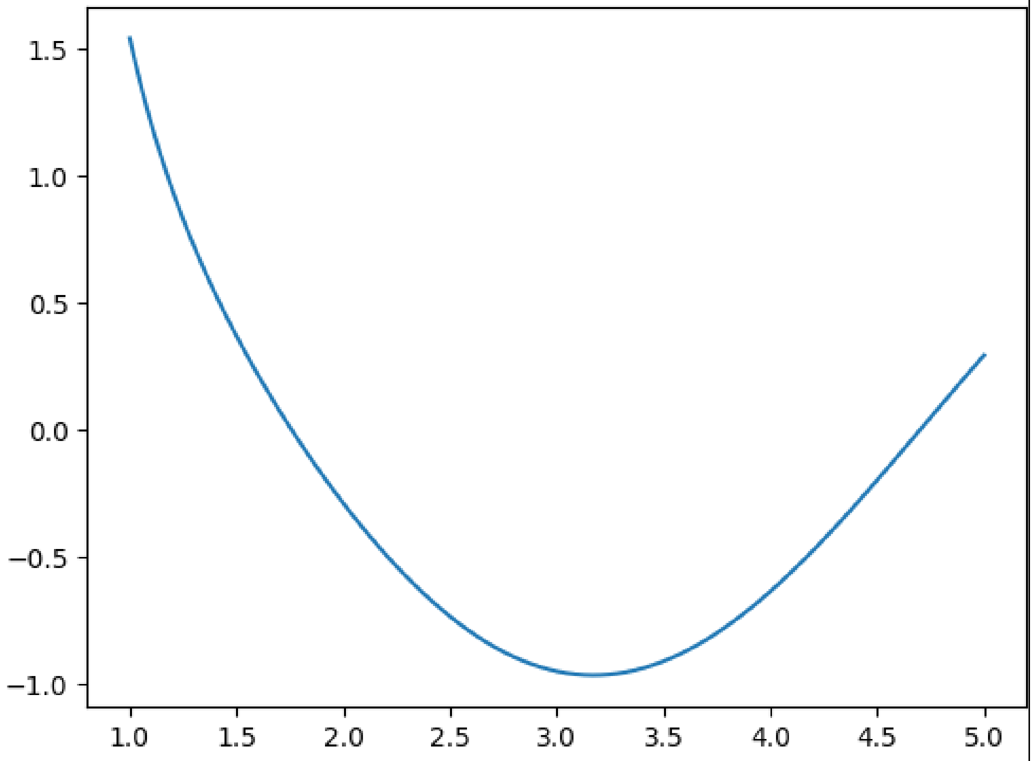
```
# Integral Eksak

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

```
# metode trapezoid

n = 1000
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 Metode Trapezoid   :", hasil)
```

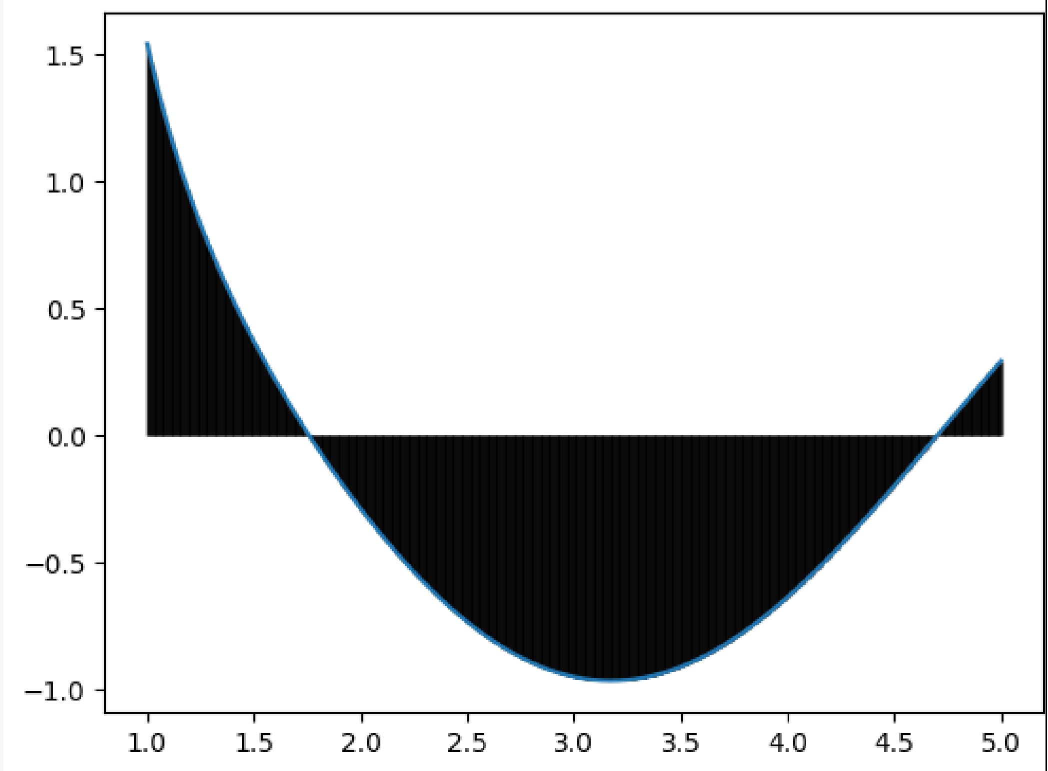
```
Hasil Metode Trapezoid   : -1.3203888525573348
```

Berdasarkan hasil integrasi metode trapezoid, maka diperoleh nilai hasil yaitu sebesar -1.3203888525573348 dimana ia mendekati nilai integral metode eksak. Adapun tahapan *plotting* serta kurva yang dihasilkan adalah sebagai berikut.

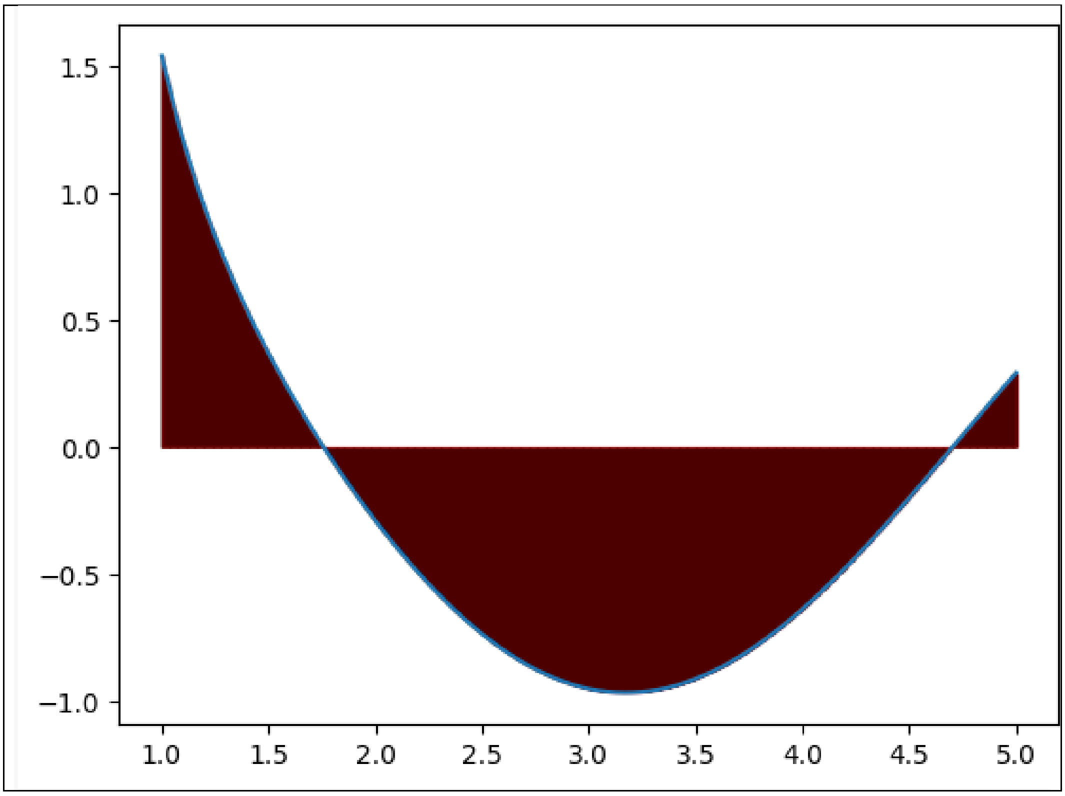
```
xp=np.linspace(a,b,1000000)
plt.plot(xp,func(xp))
plt.show()
```




```
xp=np.linspace(a,b,1000000)
plt.plot(xp,func(xp))

for i in range (n):
    plt.bar(x[i], func(x[i]), align = 'edge', width =
0.000001, edgecolor='black', alpha = 0.7)

plt.show()
```




```
xp=np.linspace(a,b,1000000)
plt.plot(xp,func(xp))

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

plt.fill_between(x, func(x), color = 'red', alpha = 0.3)
plt.show()
```



c) Metode Simpson 1/3

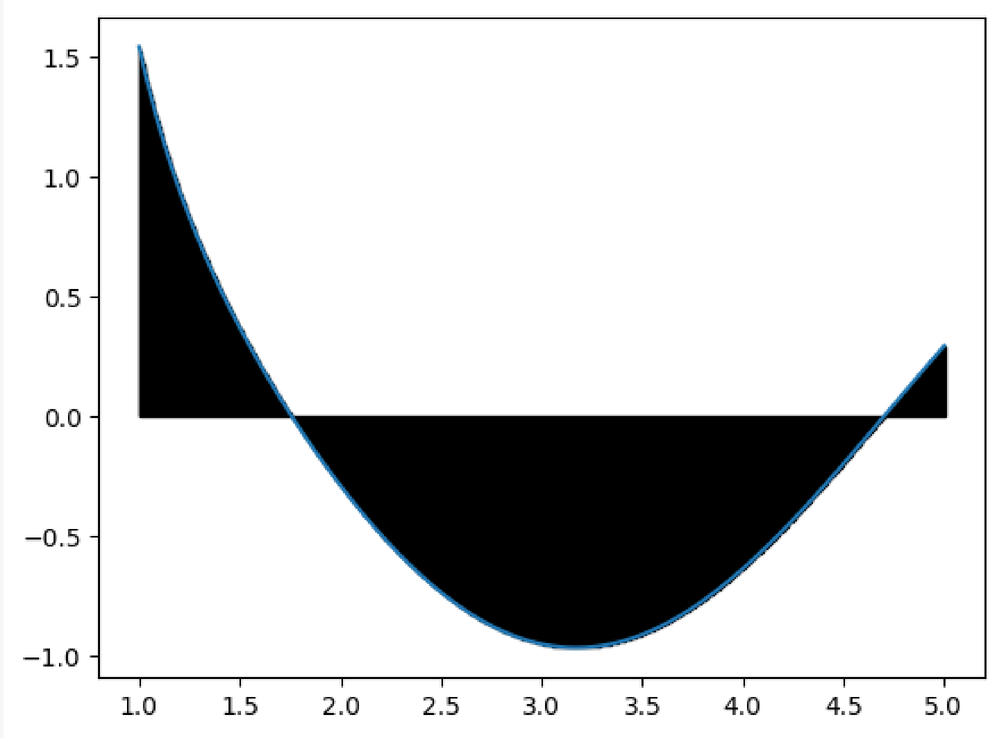
Disusun algoritma pemrograman metode integrasi Simpson 1/3 untuk mencari nilai hasil integral dari fungsi yang diberikan, kemudian melakukan *plotting* untuk menghasilkan visualisasi dari tahapan integrasi yang diberikan menggunakan metode Simpson 1/3. Adapun algoritma pemrograman serta hasil yang diperoleh ditampilkan sebagai berikut.

```
# Nama   : Gilang Pratama Putra Siswanto
# NIM    : 1227030017
# Tugas 4 Inteagral Metode Numerik - Praktikum Fisika
Komputasi

import numpy as np
import matplotlib.pyplot as plt
```

```
# Integral Eksak

def func(x):
    return ((x**(-3))+np.cos(x))
```

```
a = 1.0
b = 5.0
```

```
# Simpson Punya Aturan 1/3
if n % 2 == 0:
    n += 1

x = np.linspace(a, b, n)
dx = (x[-1] - x[0]) / (n - 1)


hasil1 = func(x[0]) + func(x[-1])

for i in range(1, n-1, 2):
    hasil1 += 4 * func(x[i])

for i in range(2, n-2, 2):
    hasil1 += 2 * func(x[i])

hasil1 *= dx / 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=dx,
color='red', edgecolor='black')
plt.show()

print("Hasil Metode Simpson 1/3    :", hasil1)
```




```
Hasil Metode Simpson 1/3    : -1.3203952593882757
```

Berdasarkan hasil integrasi metode trapezoid, maka diperoleh nilai hasil yaitu sebesar-1.3203952593882757.

BABAK 2 – Penjelasan Hasil yang Diperoleh

a) Metode eksak

Dalam menyelesaikan soal integral dari fungsi

$$\int_{1}^{5} x^{-3} + \cos(x)\ dx$$

tahapan pertama yang dilakukan adalah melakukan pemisahan dari masing-masing fungsi agar tahapan integrasi menjadi lebih terstruktur.

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

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

Setelah dilakukan pemisahan, maka dilakukan tahapan integrasi dimana integral dari fungsi x^-3 maka digunakan Langkah yang ditempuh adalah menggunakan tahapan integrasi biasa dan untuk fungsi cos (x), tahapan integrasi yang dilakukan adalah dengan menggunakan tahapan integrasi untuk trigonometri dimana hasil integrasi yang ditempuh menghasilkan hasil sebagai berikut.

$$= \int_1^5 \frac{1}{-3+1} x^{-3+1}\, dx + \int_1^5 \cos(x)\, dx$$

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

Nilai pangkat atau n yaitu -3 diletakkan di pada 1/n+1 dan pangkat x^n+1 sebagaimana tahapan integrasi biasa dan integral daripada cos(x) tidak lain adalah sin(x) kemudian keduanya dievaluasi dari batas atas 5 hingga dikurangi batas bawah 1 dengan tahapan sebagai berikut.

$$\approx -\frac{1}{2}(5)^{-2} + \frac{1}{2}(1)^{-2} + \sin(5) - \sin(1)$$

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

$$= \frac{-1}{50} + \frac{25}{50} - 0{,}958 - 0{,}841$$

$$= \frac{24}{50} - 0{,}9589 - 0{,}8414$$

$$= -1{,}3203$$

Parameter yang perlu diperhatikan adalah nilai x pada sin(x) ini ditetapkan dalam bentuk radian sehingga menghasilkan nilai negative pada sin(5) karena berada pada kuadran III dimana eksekusi dari fungsi sinus adalah negative sedangkan nilai sin(1) masih berada pada kuadran 1 sehingga bernilai positif, namun karena ia merupakan batas bawah maka operasi yang berlaku adalah pengurangan. Kemudian, dengan menjumlahkan hasil keseluruhan evaluasi ini menghasilkan nilai hasil integral sebesar -1,3203.

b) Metode Trapezoid

Dalam menyelesaikan soal integral dalam metode trapezoid, maka sebelumnya dilakukan *import* pustaka yaitu numpy untuk penggunaan fungsi matematika dalam kode program serta matplotlib untuk pembuatan grafik berdasarkan data yang diperoleh.

```
# Nama   : Gilang Pratama Putra Siswanto
# NIM    : 1227030017
# Tugas 4 Inteagral Metode Numerik - Praktikum Fisika
Komputasi

import numpy as np
import matplotlib.pyplot as plt
```

Tahapan selanjutnya yaitu melakukan pendefinisian fungsi yang tertera sesuai dengan soal dengan menentukan batas atas yaitu b = 5.0 dan batas bawah yaitu a = 1.0 sesuai dengan soal.

```
# Integral Eksak

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

Setelah didefinisikan fungsi serta batas atas dan batas bawah, Maka ditentukan berapa banyak data pengujian atau n hitung yang diiterasikan untuk menguji metode trapezoid ini. Semakin tinggi nilai n, maka iterasi semakin sempurna dan hasil yang diperoleh semakin mendekati nilai perhitungan secara eksak. Selain itu, nilai x ditentukan dengan plotting garis dari batas bawah 1 hingga batas atas 5. Ditentukan nilai n yaitu sebesar 1000. Kemudian, dicari seberapa besar nilai perubahan x dinotasikan sebagai dx. Semakin kecil nilai dx maka kemudian akan memperoleh nilai perhitungan yang semakin presisi. Kemudian, dengan mengacu pada persamaan trapezoid

$$I = \frac{h}{2}\left(f(x_0) + 2\sum_{i=1}^{n-1} f(x_i) + f(x_n) \right)$$

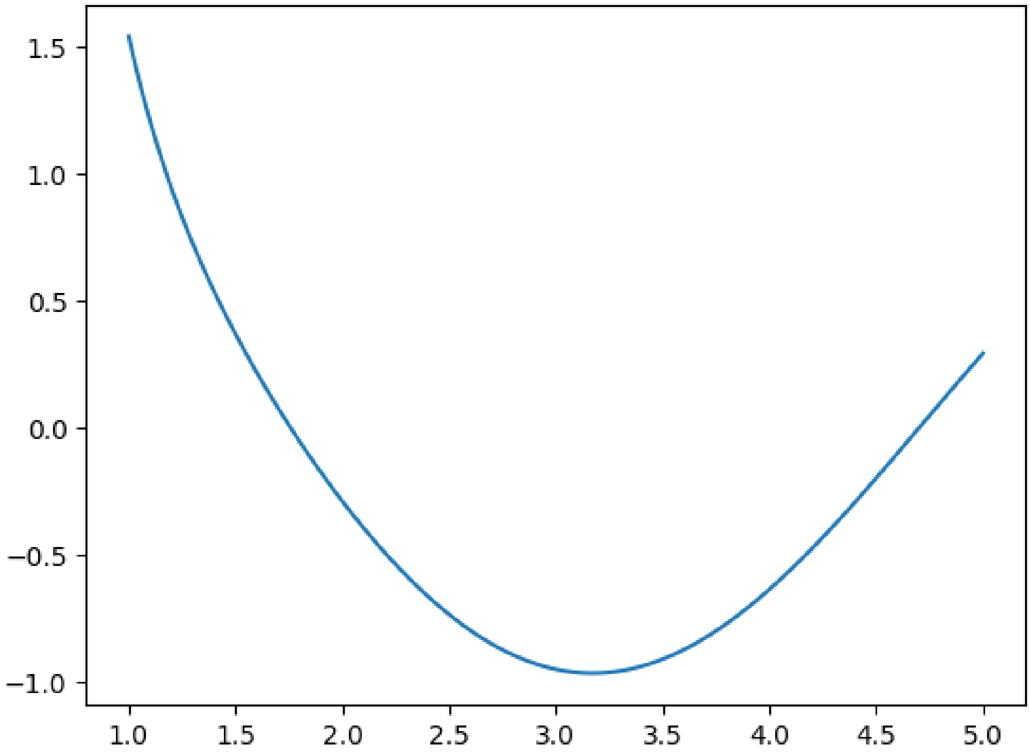
maka ditentukan penjumlahan dengan notasi sigma dari i = 1 hingga n-1 sebagai penjumlahan dari fungsi f(xi). Kemudian, diatur algoritma perhitungan nilai hasil dengan mengikuti persamaan trapezoid ini ke dalam kode program hingga kemudian melakukan pencetakan nilai hasilnya sebagaimana kode program berikut.

```
# metode trapezoid

n = 1000
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 Metode Trapezoid    :", hasil)
```

```
Hasil Metode Trapezoid    : -1.3203888525573348
```

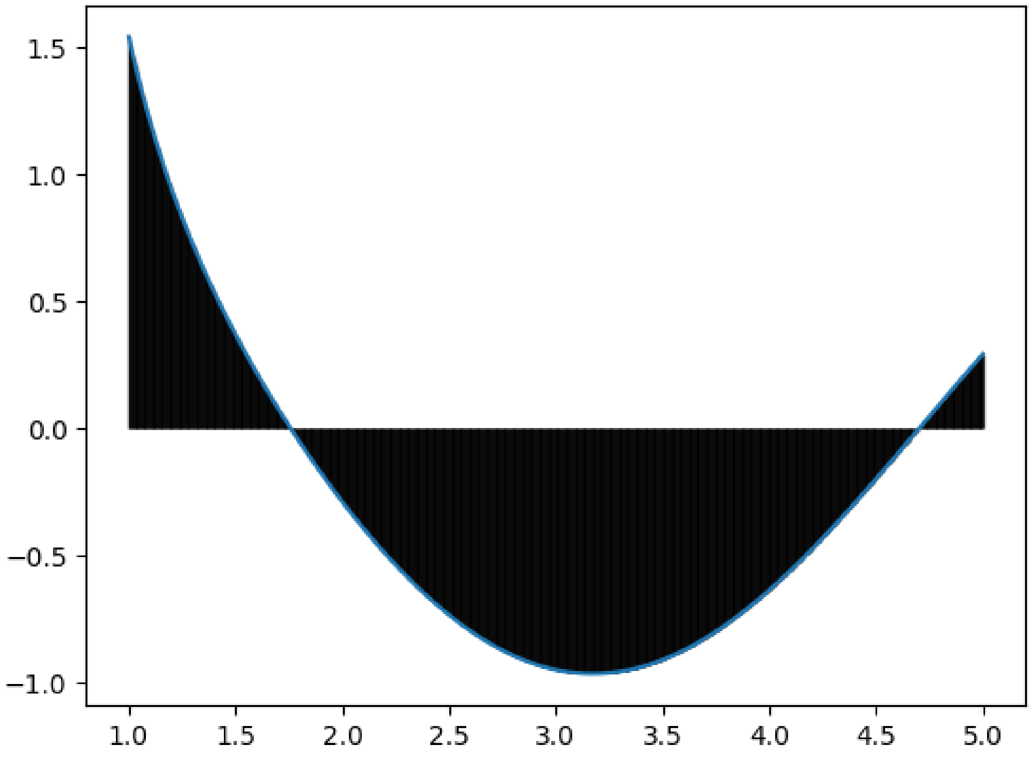
Berdasarkan nilai pengujian pada metode trapezoid, maka diperoleh nilai hasil sebesar -1.3203888525573348 dimana ia mendekati nilai perhitungan dalam metode eksak. Selain itu, untuk menggambarkan bagaimana metode trapezoid, maka dilakukan *plotting* dengan parameter sumbu x yaitu x dan sumbu y yaitu f(x) dengan parameter ketelitian sebesar 1000000. Kemudian, hasil *plotting* tersebut ditampilkan dalam grafik sebagai berikut.



Selanjutnya, untuk menggambarkan nilai tengah dari masing masing-masing trapezoid di setiap n iterasi, maka dilakukan plotting garis atau bar dengan menggunakan algoritma pemrograman sebagai berikut.

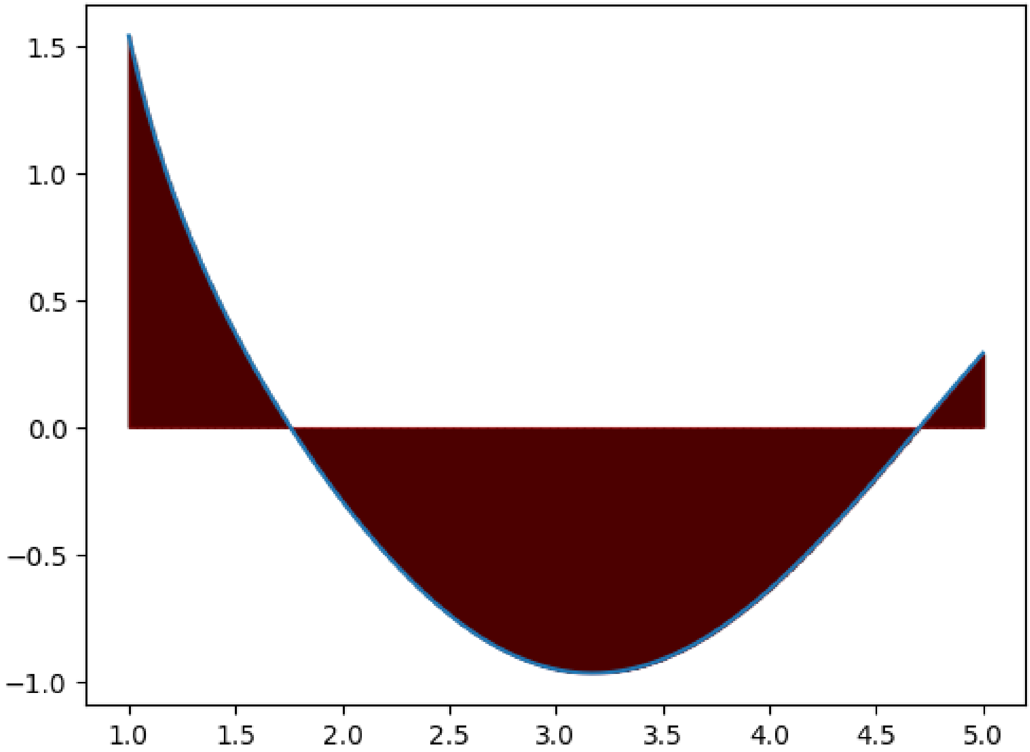
```
xp=np.linspace(a,b,1000000)
plt.plot(xp,func(xp))

for i in range (n):
    plt.bar(x[i], func(x[i]), align = 'edge', width =
0.000001, edgecolor='black', alpha = 0.7)

plt.show()
```

Kemudian, hasil *plotting* yang dihasilkan ditampilkan dalam bentuk grafik garis-garis tengah yang terdistribusi di setiap n iterasi sebagai berikut.



Dengan menurunkan opasitas garis menjadi 70%, maka dapat ditinjau bahwa terdapat garis hitam yang tersebar dari titik x = 1 hingga x = 5 sebanyak n iterasi yang digunakan yaitu sebesar 1000. Untuk mengamati bagaimana metode trapezoid mencari nilai integral dari fungsi yang diberikan, maka digunakan fungsi plt.fill_between dari matplotlib untuk mengisi warna antara titik n dengan titik n sebelumnya dengan warna merah dan opasitasnya sebesar 30%. Kemudian, teramati bahwa *plotting* trapezoid mengisi secara lebih presisi di sepanjang n dari batas bawah hingga batas atas dan cukup presisi mengisi seluruh kurva dengan minim bagian trapezoid yang tidak menutupi atau melampaui bagian di bawah kurva. Adapun kode program dan grafik yang ditampilkan dengan menggunakan fungsi trapezoid adalah sebagai berikut.

```
xp=np.linspace(a,b,1000000)
plt.plot(xp,func(xp))

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

plt.fill_between(x, func(x), color = 'red', alpha = 0.3)
plt.show()
```



c) Metode Simpson 1/3

Dalam menyelesaikan soal integral dalam metode Simpson 1/3, sama seperti tahapan sebelumnya maka yaitu dilakukan *import* pustaka yaitu numpy untuk penggunaan fungsi matematika dalam kode program serta matplotlib untuk pembuatan grafik berdasarkan data yang diperoleh.

```
# Nama    : Gilang Pratama Putra Siswanto
# NIM     : 1227030017
# Tugas 4 Inteagral Metode Numerik - Praktikum Fisika
Komputasi

import numpy as np
import matplotlib.pyplot as plt
```

Tahapan selanjutnya yaitu turut melakukan pendefinisian fungsi yang tertera sesuai dengan soal dengan menentukan batas atas yaitu b = 5.0 dan batas bawah yaitu a = 1.0 sesuai dengan soal. Dalam penggunaan piranti lunak Google Collab, algoritma ini dan sebelumnya tidak perlu didefinisikan Kembali dan cukup dijalankan ketika hendak mengeksekusi program.

```
# Integral Eksak

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

Algoritma selanjutnya yaitu untuk memberikan logika jika iterasi yang diberikan ganjil atau genap. Ketika nilai n iterasi berupa genap, makai a akan ditambahkan sehingga ia menjadi iterasi dengan jumlah ganjil. Kemudian, nilai perubahan x cukup berbeda dengan sebelumnya dimana nilai a dan b sebelumnya diganti dengan nilai x[-1] – x [0] dibagi dengan pengurangan nilai n iterasi – 1. Untuk bilangan dengan penjumlahan bilangan n adalah ganjil, maka nilai yang diberikan adalah 4 kali nilai dikali fungsi f(xi) sepanjang iterasi n hingga n-1. Untuk bilangan n adalah genap, maka operasi yang dilakukan adalah perkalian yaitu 2 kali dari nilai fungsi f(xi) sepanjang n iterasi hingga n-2. Hasil perkalian ini kemudian turut dijumlah dan kemudian nilai keseluruhan ini dibagi 3. Hal ini sejalan dengan persamaan Simpson 1/3, yaitu sebagai berikut.

$$I = \frac{h}{3}\left(f_0 + 4\sum_{i=odd}^{n-1} f_i + 2\sum_{i=even}^{n-2} f_i + f_n\right)$$

Adapun algoritma pemrograman yang disusun adalah sebagai berikut.

```
# Simpson Punya Aturan 1/3
if n % 2 == 0:
    n += 1

x = np.linspace(a, b, n)
dx = (x[-1] - x[0]) / (n - 1)
```

```
hasil1 = func(x[0]) + func(x[-1])

for i in range(1, n-1, 2):
    hasil1 += 4 * func(x[i])

for i in range(2, n-2, 2):
    hasil1 += 2 * func(x[i])

hasil1 *= dx / 3
```

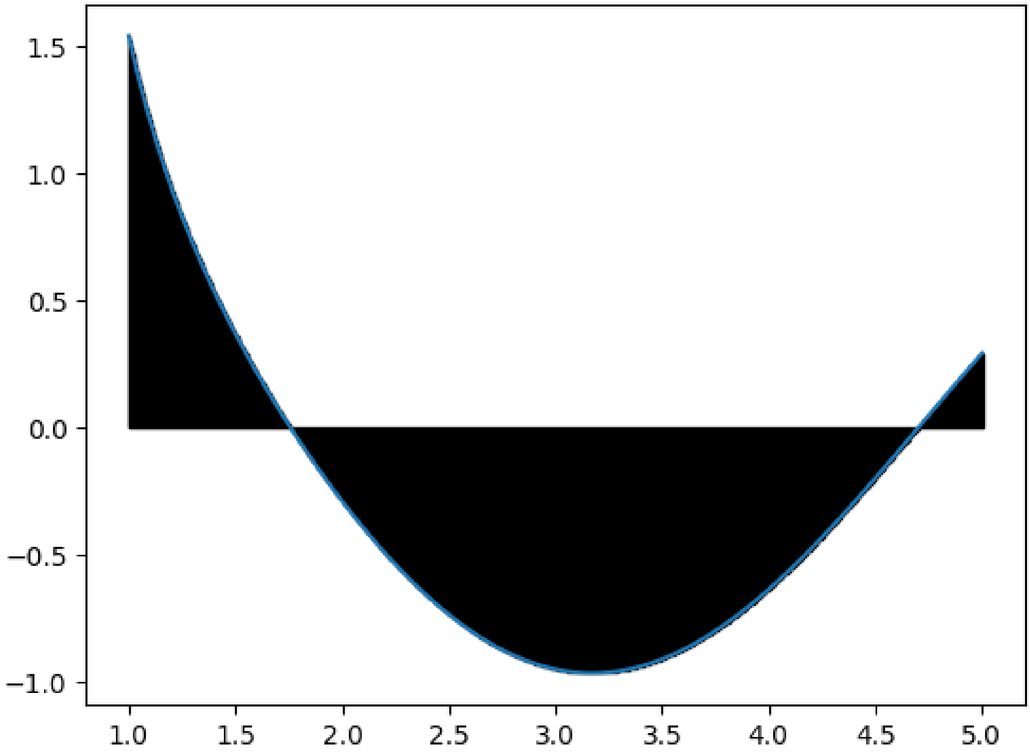
Kemudian hasil perhitungan dicetak dengan menggunakan fungsi print sebagai berikut.

```
print("Hasil Metode Simpson 1/3    :", hasil1)
```

```
Hasil Metode Simpson 1/3    : -1.3203952593882757
```

Berdasarkan hasil perhitungan, maka diperoleh nilai hasil yaitu sebesar -1.3203952593882757 dimana hasil ini mendekati nilai perhitungan yang dilakukan secara eksak. Untuk mengamati bagaimana bentuk integral yang ditampilkan melalui *plotting*, maka digunakan fungsi plt.bar untuk menggambarkan bagaimana distribusi garis di sepanjang n pengujian menggunakan metode Simpson 1/3. Adapun algoritma pemrograman yang dilakukan serta hasil grafik yang dihasilkan ditampilkan sebagai berikut.

```
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()
```



BABAK 3 – Perbedaan Tiap Metode dan Rekomendasi Metode Paling Efektif

Di antara perbedaan yang dapat diamati dari metode eksak, metode trapezoid, serta metode Simpson 1/3 yaitu dari algoritma pencarian nilai di bawah kurva serta dari hasil *plotting* yang dieksekusi. Pada metode eksak, algoritma yang ditempuh adalah menggunakan teknik kalkulus integral biasa untuk fungsi x^-3 dan menggunakan metode integral untuk trigonometri pada fungsi sin(x) kemudian mengevaluasi hasil integrasi keduanya sesuai dengan batas atas dan bawah yang ditentukan. Maka kemudian hasil integrasi ini tidak lain adalah perkalian antara nilai fungsi f(x) di sepanjang batas x (yaitu 1 hingga 5). Pada metode trapezoid, algoritma yang ditempuh adalah menggunakan penjumlahan dari trapezoid atau yang disusun di sepanjang kurva. Semakin banyak trapezoid yang ditentukan maka ia berpengaruh terhadap seberapa presisi trapezoid ini menutup luas di bawah kurva. Pada metode simpson 1/3, algoritma yang dilakukannya cenderung menggunakan penjumlahan dari fungsi yang diuji dalam n uji ganjil dan genap dengan masing-masing fungsi tersebut dikali 4 untuk n ganjil dan dikali 2 untuk n bilangan genap. Sesuai dengan namanya, maka hasil penjumlahan ini kemudian dibagi 3 sehingga parameter utama yang menjadi pertimbangan adalah metode Simpson ini cenderung bergantung pada tahapan mengubah fungsi dalam bentuk ganjil.

Dengan melihat seberapa presisi metode yang ditempuh untuk mencari luas bidang di bawah kurva, maka metode yang cenderung tepat adalah metode Simpson 1/3. Dengan jumlah n uji yang sama yaitu 1000, maka bidang yang tertutupi di bawah kurva lebih tebal dan hasil yang diperoleh cenderung presisi jika tahapan integrasi ditempuh menggunakan metode Simpson 1/3. Namun, penggunaan metode ini turut memerlukan beberapa pertimbangan dimana ia harus menggunakan beberapa algoritma seperti melakukan penjumlahan fungsi pada n ganjil dan genap, kemudian hasil penjumlahannya dibagi 3 sehingga algoritma yang ditempuh cukup kompleks.