Nama : Gina Nabila
NIM : 21120122130055
Mata Kuliah : Metode Numerik – Kelas D
Github Link : https://github.com/gin-na/Integrasi-Trapezoid_Metode-Numerik_Gina-Nabila

Implementasi Integrasi Numerik untuk Menghitung Estimasi nilai Pi

Nilai pi dapat dihitung secara numerik dengan mencari nilai integral dari fungsi $f(x) = \frac{4}{(1 + x^2)}$ dari 0 sampai 1.

Diinginkan implementasi penghitungan nilai integral fungsi tersebut secara numerik dengan metode:

1. integrasi Reimann (Metode 1)
2. Integrasi trapezoid (Metode 2)
3. Integrasi Simpson 1/3 (Metode 3)
4. Tugas mahasiswa:

Mahasiswa membuat kode sumber dengan bahasa pemrograman yang dikuasai untuk mengimplementasikan solusi di atas, dengan ketentuan:

Dua digit NIM terakhir % 3 = 0 mengerjakan dengan Metode 1

Dua digit NIM terakhir % 3 = 1 mengerjakan dengan Metode 2

Dua digit NIM terakhir % 3 = 2 mengerjakan dengan Metode 3

Sertakan kode testing untuk menguji kode sumber tersebut untuk menyelesaikan problem dengan ketentuan sebagai berikut:

Menggunakan variasi nilai N = 10, 100, 1000, 10000

Hitung galat RMS dan ukur waktu eksekusi dari tiap variasi N. Nilai referensi pi yang digunakan adalah 3.14159265358979323846

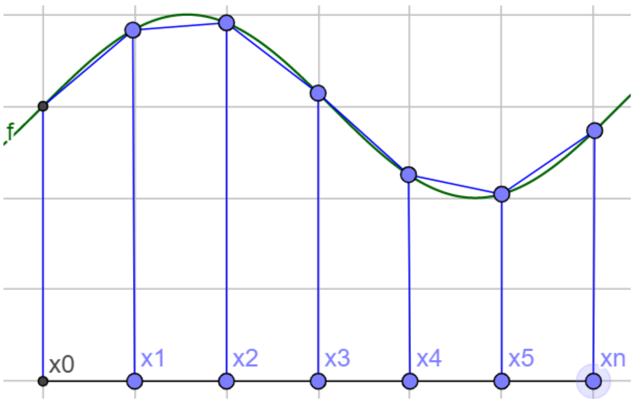
Mengunggah kode sumber tersebut ke Github dan setel sebagai publik. Berikan deskripsi yang memadai dari project tersebut. Masukkan juga dataset dan data hasil di repositori tersebut.

Buat dokumen docx dan pdf yang menjelaskan alur kode dari (1), analisis hasil, dan penjabarannya. Sistematika dokumen: Ringkasan, Konsep, Implementasi Kode, Hasil Pengujian, dan Analisis Hasil. Analisis hasil harus mengaitkan antara hasil, galat, dan waktu eksekusi terhadap besar nilai N.

Integrasi Trapezoid

1. Ringkasan Penyelesaian dengan Metode Integrasi Trapezoid

Integrasi trapezoid adalah salah satu metode numerik untuk menghitung integral dari fungsi dengan mempartisi suatu fungsi menjadi n bagian lalu membentuk partisi-partisi tersebut sedemikian rupa sehingga menjadi bentuk trapesium.



Gambar 1. 1 Ilustrasi Integral dengan Metode Trapezoid

Ilustrasi di atas adalah mempartisi suatu fungsi menjadi n bagian. Luas di bawah kurva mulai dari x_0 hingga x_n dapat ditulis sebagai berikut:

$$\int_a^b f(x)dx$$

Dimana $a = x_0$ dan $b = x_n$

Luas tersebut dapat diaproksimasi secara numerik dengan meghitung luas setiap trapesium. Luas trapesium adalah $\frac{1}{2} \times \text{tinggi} \times \text{jumlahsisisejajar}$. Sehingga h untuk setiap trapesium bernilai sama yaitu $h = \frac{(b-a)}{n}$.

Sehingga, secara ringkas aproksimasi integral dengan metode trapezoid adalah:

$$\int_a^b f(x)dx \approx \frac{h}{2}\left[Y0 + 2\sum_{i=1}^{N-1} Yi + YN\right]$$

2. Konsep Algoritma Metode Trapezoid:

- Mendefinisikan $y = f(x)$
- Menentukan batas Bawah (a) dan batas atas integrasi (b)
- Menentukan jumlah pembagi n
- Menghitung $h = \frac{b-a}{n}$
- Menghitung $L = \frac{1}{2}(f_0 + 2\sum_{i=1}^{n-1} f_i + f_n)$

3. Implementasi Kode dan Penjelasan Alur Kode

```
import numpy as np
import time

# Fungsi dari soal yang akan diintegrasikan
def fungsi(x):
    return 4 / (1 + x**2)

# Fungsi penyelesaian dengan metode integrasi trapezoid
def integrasi_trapezoid(f, a, b, N):
    x = np.linspace(a, b, N+1)
    y = f(x)
    h = (b - a) / N
    integral = (h / 2) * (y[0] + 2 * np.sum(y[1:-1]) + y[-1])
    return integral

# Nilai referensi pi
nilai_pi = 3.14159265358979323846

# Variasi nilai N yang digunakan
nilai_N = [10, 100, 1000, 10000]
```

```
# Hasil penyelesaian
for N in nilai_N:
    waktu_mulai = time.time()
    estimasi_pi = integrasi_trapezoid(fungsi, 0, 1, N)
    waktu_selesai = time.time()

    # Menghitung galat RMS
    galat_rms = np.sqrt((estimasi_pi - nilai_pi)**2)

    # Menghitung waktu eksekusi
    waktu_eksekusi = waktu_selesai - waktu_mulai

    # Menampilkan hasil
    print(f"N = {N}")
    print(f"Estimasi nilai pi = {estimasi_pi}")
    print(f"Galat RMS = {galat_rms}")
    print(f"Waktu eksekusi = {waktu_eksekusi} detik\n")
```

Penjelasan alur perhitungan dalam kode yaitu sebagai berikut:

a) Inisiasi fungsi dan parameter

```
# Fungsi dari soal yang akan diintegrasikan
def fungsi(x):
    return 4 / (1 + x**2)

# Fungsi penyelesaian dengan metode integrasi trapezoid
def integrasi_trapezoid(f, a, b, N):
    x = np.linspace(a, b, N+1)
    y = f(x)
    h = (b - a) / N
    integral = (h / 2) * (y[0] + 2 * np.sum(y[1:-1]) + y[-1])
    return integral
```

Pada kode di atas didefinisikan fungsi yang akan diintegrasikan yaitu $f(x) = \frac{4}{1+x^2}$ kemudian fungsi integrase trapezoid digunakan untuk menghitung integral dari `f` pada interval `[a, b]` menggunakan metode trapezoid dengan `N` subintervals.

b) Inisiasi variabel yang digunakan

```
# Nilai referensi pi
nilai_pi = 3.14159265358979323846

# Variasi nilai N yang digunakan
nilai_N = [10, 100, 1000, 10000]
```

Diberikan nilai referensi pi yang digunakan untuk menghitung galat RMS serta daftar nilai N yang akan digunakan untuk menghitung integral.

c) Proses perhitungan

```
# Hasil penyelesaian
for N in nilai_N:
    waktu_mulai = time.time()
    estimasi_pi = integrasi_trapezoid(fungsi, 0, 1, N)
    waktu_selesai = time.time()

    # Menghitung galat RMS
    galat_rms = np.sqrt((estimasi_pi - nilai_pi)**2)

    # Menghitung waktu eksekusi
    waktu_eksekusi = waktu_selesai - waktu_mulai
```

Untuk setiap nilai N dalam `nilai_N`, dilakukan langkah-langkah perhitungan untuk menghitung estimasi pi, waktu eksekusi, dan galat RMS.

d) Hasil yang ditampilkan

```
# Menampilkan hasil
print(f"N = {N}")
print(f"Estimasi nilai pi = {estimasi_pi}")
print(f"Galat RMS = {galat_rms}")
print(f"Waktu eksekusi = {waktu_eksekusi} detik\n")
```

Hasil dari perhitungan dicetak menggunakan kode di atas.

4. Hasil Pengujian

```
N = 10
Estimasi nilai pi = 3.1399259889071587
Galat RMS = 0.0016666646826344333
Waktu eksekusi = 0.0004153251647949219 detik

N = 100
Estimasi nilai pi = 3.141575986923129
Galat RMS = 1.6666666664111318e-05
Waktu eksekusi = 0.00018739700317382812 detik

N = 1000
Estimasi nilai pi = 3.141592486923127
Galat RMS = 1.6666666624587378e-07
Waktu eksekusi = 0.00018453598022460938 detik

N = 10000
Estimasi nilai pi = 3.1415926519231263
Galat RMS = 1.666666804567285e-09
Waktu eksekusi = 0.000270843505859375 detik
```

5. Analisis Hasil

Hasil yang dicetak berisi estimasi nilai pi, galat RMS, dan waktu eksekusi. Nilai pi dihasilkan berdasarkan perhitungan menggunakan metode integrase trapezoid. Ralat RMS didapatkan dari rumus $\sqrt{(Estimate - Reference)^2}$. Sedangkan waktu eksekusi dihasilkan berdasarkan perhitungan waktu selesai dikurangi waktu mulai.

Hasil yang dihasilkan saling berkaitan satu sama lain. Dengan bertambahnya nilai N, estimasi nilai pi seharusnya semakin mendekati nilai sebenarnya π, karena metode trapezoid menjadi lebih akurat dengan lebih banyak subintervals. Nilai galat RMS umumnya akan berkurang dengan bertambahnya nilai N, hal ini menunjukkan bahwa estimasi semakin akurat. Sedangkan waktu eksekusi biasanya akan meningkat dengan bertambahnya nilai N karena ada lebih banyak perhitungan yang harus dilakukan.

Sehingga, dapat disimpulkan bahwa nilai N yang lebih besar cenderung menghasilkan estimasi yang lebih akurat untuk nilai pi dengan galat RMS yang lebih kecil, tetapi dengan waktu eksekusi yang cenderung lebih lama.