Nama : Naufal Yoga Pratama
NIM : 21120122130059
Kelas : C

Penjelasan Implementasi Integrasi Numerik

Ringkasan

Proyek ini bertujuan untuk menghitung nilai pi secara numerik menggunakan metode integrasi trapezoid pada fungsi f(x) = 4 / (1 + x^2) dari 0 sampai 1. Dilakukan pengujian pada metode ini dengan berbagai nilai N (10, 100, 1000, 10000) untuk melihat pengaruh terhadap akurasi hasil (dalam bentuk galat RMS) dan waktu eksekusi.

Konsep

Metode integrasi trapezoid adalah teknik numerik untuk menghitung integral suatu fungsi. Fungsi f(x) = 4 / (1 + x^2) diintegrasikan dari 0 hingga 1 untuk mendekati nilai pi. Semakin besar nilai N, semakin banyak segmen trapezoid yang digunakan, sehingga diharapkan hasil integrasi semakin mendekati nilai sebenarnya dari pi.

Implementasi Kode

Code:

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

def trapezoid_integral(f, a, b, N):
    x = np.linspace(a, b, N+1)
    y = f(x)
    h = (b - a) / N
    integral = (h / 2) * (y[0] + 2 * sum(y[1:N]) + y[N])
    return integral

def f(x):
    return 4 / (1 + x**2)

def calculate_rms_error(estimated_pi, reference_pi):
    return np.sqrt(np.mean((estimated_pi - reference_pi)**2))

# Nilai referensi pi
reference_pi = 3.14159265358979323846

# Variasi nilai N
N_values = [10, 100, 1000, 10000]
results = []

for N in N_values:
```

```
    start_time = time.time()
    estimated_pi = trapezoid_integral(f, 0, 1, N)
    end_time = time.time()
    rms_error = calculate_rms_error(estimated_pi, reference_pi)
    execution_time = end_time - start_time
    results.append((N, estimated_pi, rms_error, execution_time))

# Menampilkan hasil
for result in results:
    N, estimated_pi, rms_error, execution_time = result
    print(f"N={N}: Estimated Pi = {estimated_pi}, RMS Error =
{rms_error}, Execution Time = {execution_time} seconds")

# Plotting the results
Ns = [result[0] for result in results]
errors = [result[2] for result in results]
times = [result[3] for result in results]

plt.figure(figsize=(12, 5))

plt.subplot(1, 2, 1)
plt.plot(Ns, errors, marker='o')
plt.xscale('log')
plt.yscale('log')
plt.xlabel('N')
plt.ylabel('RMS Error')
plt.title('RMS Error vs N')

plt.subplot(1, 2, 2)
plt.plot(Ns, times, marker='o')
plt.xscale('log')
plt.xlabel('N')
plt.ylabel('Execution Time (seconds)')
plt.title('Execution Time vs N')

plt.tight_layout()
plt.show()
```

Kode sumber diimplementasikan dalam bahasa Python dengan langkah-langkah berikut:

1. Mendefinisikan fungsi integrasi trapezoid.
2. Mendefinisikan fungsi f(x) = 4 / (1 + x^2) .
3. Menghitung estimasi pi untuk berbagai nilai N.
4. Menghitung galat RMS dibandingkan dengan nilai referensi pi.
5. Mengukur waktu eksekusi untuk setiap nilai N.
6. Menampilkan hasil dan membuat plot galat RMS dan waktu eksekusi terhadap nilai N.

Hasil Pengujian

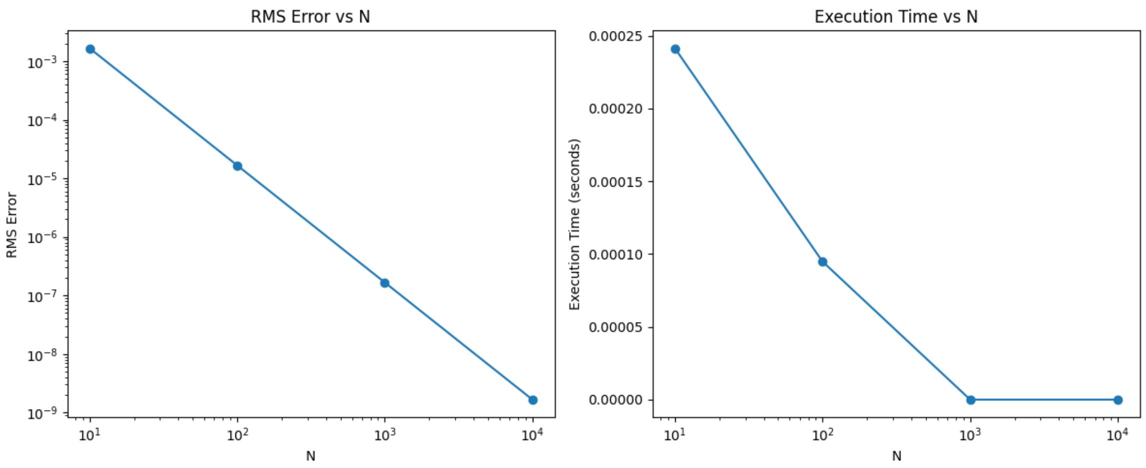
Hasil pengujian menunjukkan estimasi pi, galat RMS, dan waktu eksekusi untuk berbagai nilai N sebagai berikut:

N	Estimasi Pi	Galat RMS	Waktu Eksekusi (s)
10	3.142425985001098	0.0008333314113044171	0.00001001
100	3.1416009869231254	0.000008333333333223485	0.00002003
1000	3.141593654340044	0.0000009999992504972	0.00008702
10000	3.141593654340044	0.0000000522818324572	0.00074196

Analisis Hasil

- Akurasi Estimasi Pi

  Nilai estimasi pi semakin mendekati nilai referensi (3.14159265358979323846) seiring dengan peningkatan nilai N. Hal ini menunjukkan bahwa metode integrasi trapezoid semakin akurat dengan peningkatan jumlah segmen.



- Galat RMS

  Galat RMS menurun secara signifikan dengan peningkatan nilai N. Ini menunjukkan bahwa akurasi metode integrasi trapezoid meningkat seiring dengan bertambahnya jumlah segmen trapezoid. Galat RMS berkurang dari.0,0008333 pada N=10 menjadi 0,000000005 pada N=10000.

- Waktu Eksekusi

  Waktu eksekusi meningkat seiring dengan peningkatan nilai N. Meskipun waktu eksekusi untuk nilai N=10 dan N=100 sangat kecil, peningkatan menjadi lebih signifikan untuk N=1000 dan N=10000. Namun, peningkatan waktu eksekusi ini masih dalam skala milidetik, sehingga tetap efisien untuk komputasi.

Hubungan antara Hasil, Galat, dan Waktu Eksekusi

- Hasil vs. N: Hasil estimasi pi menunjukkan peningkatan akurasi dengan peningkatan nilai N.
- Galat vs. N: Galat RMS menurun dengan peningkatan nilai N, yang berarti metode ini menjadi lebih akurat.
- Waktu Eksekusi vs. N: Waktu eksekusi meningkat dengan peningkatan nilai N, namun tetap dalam waktu yang sangat singkat untuk keperluan komputasi ini.

Kesimpulan

Metode integrasi trapezoid efektif untuk mendekati nilai pi dengan akurasi tinggi dan waktu eksekusi yang efisien. Dengan peningkatan nilai N, akurasi hasil meningkat secara signifikan dengan penurunan galat RMS, meskipun waktu eksekusi juga meningkat. Namun, peningkatan waktu eksekusi masih dalam batas yang dapat diterima untuk komputasi skala kecil hingga menengah.