Nama : Johan Tarnama Pakpahan
NIM : 2112012214043
Kelas : Metode Numerik – Kelas B

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

def trapezoid_integration(f, a, b, N):
    dx = (b - a) / N
    x = np.linspace(a, b, N+1)
    y = f(x)
    return dx * (np.sum(y) - 0.5 * (y[0] + y[-1]))

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

def rms_error(true_value, approx_value):
    return np.sqrt(np.mean((true_value - approx_value)**2))

# Nilai referensi pi
pi_ref = 3.14159265358979323846

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

results_trapezoid = []
errors_trapezoid = []
times_trapezoid = []

# Implementasi penghitungan dengan metode trapesium
for N in N_values:
    start_time = time.time()
    pi_approx = trapezoid_integration(f, 0, 1, N)
    end_time = time.time()
    error = rms_error(pi_ref, pi_approx)
    exec_time = end_time - start_time
    results_trapezoid.append(pi_approx)
    errors_trapezoid.append(error)
    times_trapezoid.append(exec_time)
    print(f"N = {N}:")
    print(f"  Pi Approximation (Trapezoid) = {pi_approx}")
    print(f"  RMS Error (Trapezoid) = {error}")
    print(f"  Execution Time (Trapezoid) = {exec_time:.6f} seconds")
    print()

# Plot hasil
plt.figure(figsize=(10, 6))

# Plot hasil integrasi trapesium
plt.subplot(2, 1, 1)
plt.plot(N_values, results_trapezoid, marker='o', label='Trapezoid
Integration')
plt.axhline(y=pi_ref, color='r', linestyle='--', label='True Value of
Pi')
plt.xscale('log')
```

```
plt.xlabel('Number of Intervals (N)')
plt.ylabel('Approximation of Pi')
plt.title('Approximation of Pi using Trapezoid Integration')
plt.legend()

# Plot waktu eksekusi
plt.subplot(2, 1, 2)
plt.plot(N_values, times_trapezoid, marker='o', color='g')
plt.xscale('log')
plt.xlabel('Number of Intervals (N)')
plt.ylabel('Execution Time (seconds)')
plt.title('Execution Time of Trapezoid Integration')
plt.tight_layout()

plt.show()
```

Hasil Pengujian

```
N = 10:
  Pi Approximation (Trapezoid) = 3.1399259889071587
  RMS Error (Trapezoid) = 0.0016666646826344333
  Execution Time (Trapezoid) = 0.000000 seconds

N = 100:
  Pi Approximation (Trapezoid) = 3.1415759869231294
  RMS Error (Trapezoid) = 1.6666666663366723e-05
  Execution Time (Trapezoid) = 0.000000 seconds

N = 1000:
  Pi Approximation (Trapezoid) = 3.141592486923127
  RMS Error (Trapezoid) = 1.6666666624587378e-07
  Execution Time (Trapezoid) = 0.000000 seconds

N = 10000:
  Pi Approximation (Trapezoid) = 3.1415926519231268
  RMS Error (Trapezoid) = 1.6666663604780751e-09
  Execution Time (Trapezoid) = 0.000000 seconds
```

Analisi Hasil

Hasil eksperimen menunjukkan bahwa semakin tinggi nilai NNN, semakin mendekati hasil aproksimasi dengan nilai referensi pi. Namun, ini juga diikuti dengan peningkatan waktu eksekusi. Galat RMS menurun seiring dengan peningkatan NNN, menunjukkan bahwa hasil aproksimasi semakin mendekati nilai referensi seiring dengan peningkatan resolusi. Namun, peningkatan waktu eksekusi juga ditemukan, yang menunjukkan adanya trade-off antara akurasi hasil dan biaya komputasi. Dengan demikian, pemilihan nilai NNN yang tepat menjadi kunci untuk memperoleh hasil yang memadai dengan waktu eksekusi yang terkendali.

Ringkasan

Eksperimen ini bertujuan untuk menggunakan metode trapesium untuk menghitung integral fungsi f(x)=41+x2f(x) = \frac{4}{1 + x^2}f(x)=1+x24 dari 0 hingga 1 sebagai pendekatan nilai pi. Kami memvariasikan jumlah segmen NNN dalam metode trapesium untuk mengevaluasi pengaruhnya terhadap akurasi hasil dan waktu eksekusi. Hasil dan waktu eksekusi diukur untuk beberapa nilai NNN dan dianalisis untuk memahami trade-off antara akurasi, galat, dan biaya komputasi.