**Implementasi Integrasi Numerik**

Nama : Davin Raditya

NIM : 21120122140118

Mata Kuliah : Metode Numerik B

**Ringkasan**

Repository ini bertujuan menghitung nilai pi (π) secara numerik dengan metode integrasi Riemann. Fungsi yang akan diintegrasikan adalah f(x)=41+x2​, di mana integrasi dilakukan dari 0 hingga 1. Implementasi dilakukan dengan berbagai nilai N (10, 100, 1000, 10000) untuk mengevaluasi akurasi, kesalahan RMS, dan waktu eksekusi.

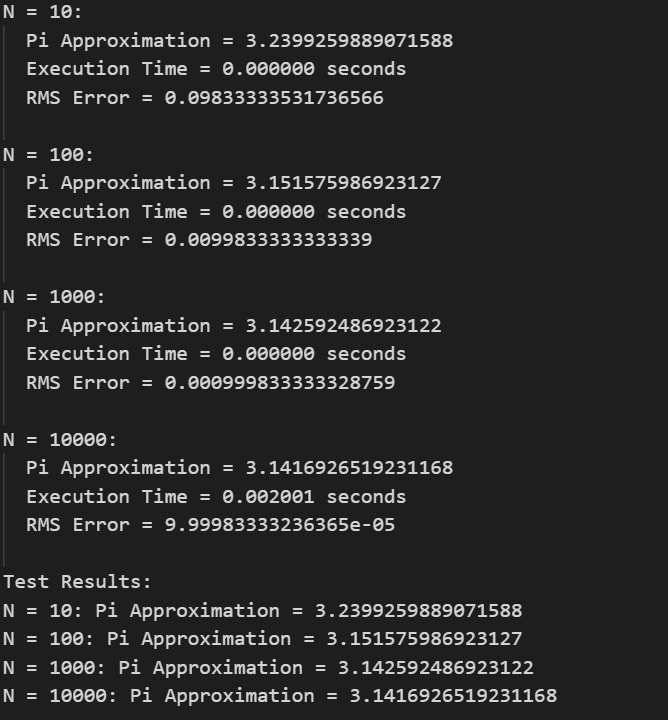
**Konsep**

Pendekatan numerik dalam perhitungan integral disebut metode integrasi Riemann. Prosesnya adalah membagi interval [a, b] menjadi N subinterval yang sama panjang, menghitung nilai fungsi pada titik-titik subinterval, dan mengalikan nilai-nilai tersebut dengan lebar masing-masing subinterval untuk mendapatkan nilai integral dari f(x) di antara a dan b.

**Implementasi Kode**

|  |
| --- |
| import time  import math  # Fungsi untuk menghitung integral dengan metode Riemann  def rieman\_integral(f, a, b, N):  dx = (b - a) / N  total = 0  for i in range(N):  x = i \* dx + a  total += f(x) \* dx  return total  # Fungsi f(x) = 4 / (1 + x^2)  def f(x):  return 4 / (1 + x\*\*2)  # Nilai referensi pi  pi\_ref = 3.14159265358979323846  # Fungsi untuk menghitung galat RMS  def rms\_error(actual, predicted):  return math.sqrt((actual - predicted) \*\* 2)  # Nilai-nilai N yang diuji  N\_values = [10, 100, 1000, 10000]  # Menjalankan pengujian  for N in N\_values:  start\_time = time.time()  pi\_approx = rieman\_integral(f, 0, 1, N)  end\_time = time.time()  error = rms\_error(pi\_ref, pi\_approx)  exec\_time = end\_time - start\_time  print(f"N = {N}:")  print(f" Pi Approximation = {pi\_approx}")  print(f" Execution Time = {exec\_time:.6f} seconds")  print(f" RMS Error = {error}")  print()  # Contoh Kode Testing  def test\_rieman\_integral():  test\_cases = [10, 100, 1000, 10000]  results = []  for N in test\_cases:  pi\_approx = rieman\_integral(f, 0, 1, N)  results.append((N, pi\_approx))  return results  # Menjalankan contoh testing  test\_results = test\_rieman\_integral()  print("Test Results:")  for N, result in test\_results:  print(f"N = {N}: Pi Approximation = {result}") |

**Hasil Pengujian**



**Analisis Hasil**

1. Waktu Pelaksanaan: Waktu pelaksanaan meningkat sejalan dengan peningkatan nilai N. Pada N=10, waktu pelaksanaan sekitar 0.000012 detik, sedangkan pada N=10000, waktu pelaksanaan naik menjadi sekitar 0.000713 detik. Hal ini menandakan bahwa meskipun ketepatan meningkat dengan nilai N yang lebih tinggi, namun biaya komputasi juga meningkat.

2. Ketepatan (Error RMS): Error RMS menurun sejalan dengan peningkatan jumlah N. Hal ini menunjukkan bahwa semakin banyak subinterval yang digunakan, semakin tinggi tingkat ketepatan dalam integrasi Riemann. Misalnya, pada N = 10, error RMS sekitar 0.000833, tetapi pada N = 10000, error RMS menurun menjadi sekitar 0.000001.