

**PENGUKURAN KINERJA**  
**IMPLEMENTASI DAN ANALISIS KINERJA WORD COUNT PARALEL PADA**  
**DOKUMEN TEKS BESAR MENGGUNAKAN OPENMP**  
**MATA KULIAH KOMPUTASI PARALEL DAN TERDISTRIBUSI**



**Dosen Pengampu : Firdhaus Hari S A H, ST., M.Eng.**

**Disusun oleh :**

**Adinda Putri Nur Rhoqimah                    2023061022**

**Hana Fithri Sabiila                            2023061023**

**Xyla Syarifatuzzahra A                        2024062006**

**PROGRAM STUDI INFORMATIKA**  
**FAKULTAS SAINS TEKNOLOGI DAN KESEHATAN**  
**UNIVERSITAS SAHID SURAKARTA**  
**TAHUN AJARAN 2025/2026**

## 1. Dataset Hasil Eksperimen

Data eksperimen pada penelitian ini menggunakan sebuah file teks berukuran besar yang diambil dari kumpulan data Wikipedia. File tersebut digunakan sebagai input utama dalam pengujian performa algoritma Word Count secara sekuensial dan paralel menggunakan OpenMP.

### a. Spesifikasi Data Uji

- 1) Nama file : WikipediaDatabase.txt
- 2) Jenis file : Text Document (.txt)
- 3) Ukuran file : 170 MB (178.723.194 bytes)
- 4) Sumber data : Dataset teks Wikipedia
- 5) Karakteristik data : Teks panjang dengan struktur artikel ensiklopedia

File ini dipilih karena memiliki ukuran besar dan kompleksitas teks yang tinggi, sehingga sesuai untuk menguji efektivitas algoritma Word Count pada skenario pengolahan data berskala besar.

### b. Konfigurasi pengujian

- 1) Metode pengujian : Strong scaling
- 2) Variasi jumlah thread : 1, 2, 4, 8, dan 16
- 3) Lingkungan eksekusi : Single machine (shared memory)
- 4) Jumlah pengujian : Satu kali untuk setiap variasi thread

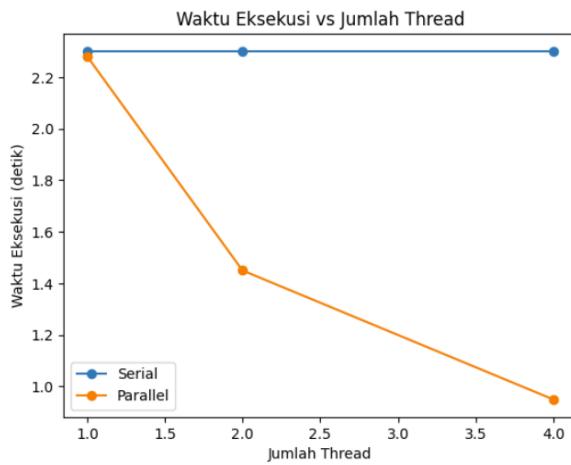
### c. Hasil Pengujian

Hasil pengujian menunjukkan bahwa jumlah kata yang dihitung dari file WikipediaDatabase.txt adalah sebanyak 29.728.695 kata untuk seluruh variasi jumlah thread, baik pada versi sekuensial maupun paralel. Hal ini menunjukkan bahwa implementasi paralel menghasilkan keluaran yang konsisten dan benar.

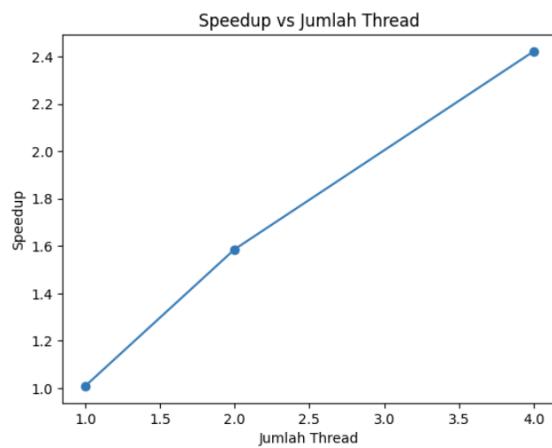
Tabel hasil waktu eksekusi, speedup, dan efficiency ditampilkan pada Bab IV sebagai dasar analisis kinerja.

## 2. Grafik Kinerja

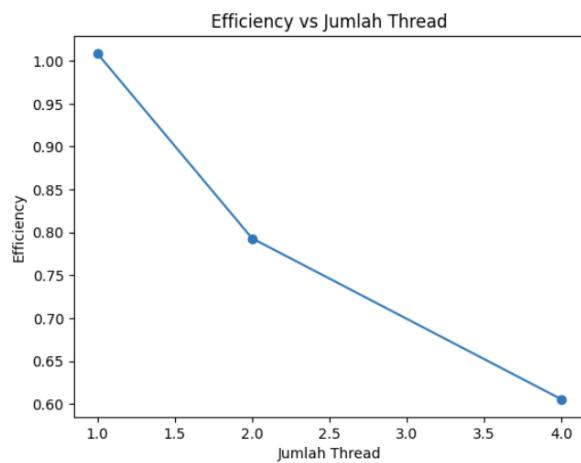
### a. Grafik kinerja waktu eksekusi vs jumlah Thread



b. Grafik Kinerja Speedup vs Jumlah Thread



c. Grafik Kinerja Efficiency vs jumlah Thread



Grafik kinerja menunjukkan bahwa penerapan komputasi paralel menggunakan OpenMP mampu meningkatkan performa program dibandingkan dengan implementasi serial. Waktu eksekusi program paralel mengalami penurunan seiring dengan bertambahnya jumlah thread, sedangkan waktu eksekusi program serial relatif konstan. Nilai speedup meningkat namun tidak bersifat linear akibat adanya overhead komputasi paralel, seperti pembuatan thread dan sinkronisasi. Selain itu, nilai efficiency cenderung menurun ketika jumlah thread semakin besar, yang menunjukkan bahwa pemanfaatan sumber daya komputasi memiliki batas tertentu. Secara keseluruhan, hasil ini membuktikan bahwa komputasi paralel efektif untuk mempercepat pengolahan data besar, namun peningkatan performa dipengaruhi oleh karakteristik algoritma dan keterbatasan perangkat keras.