

Challenges:

- **Thread Management:** Using multiple threads or tasks introduces synchronization overhead. Poor management might lead to performance degradation rather than improvement.
- **Race Conditions and Deadlocks:** With concurrent access to shared resources, ensuring thread safety through locks or other synchronization mechanisms.

Tasks:

- **Filtering Stage:** Identify how to partition scanlines or groups of scanlines for parallel processing.
- **Compression Stage:** Determine strategies for breaking the deflate compression into independent blocks to apply parallelization.
- **Task Scheduling:** Create mechanisms to schedule tasks efficiently, ensuring balanced load distribution.
- **Scanline Parsing:** Implement functions to read and prepare scanlines in a thread-safe manner.
- **Benchmarking:** Use profiling tools to measure performance improvements and identify bottlenecks.
- **Tuning:** Experiment with different block sizes, thread counts, and task granularities to maximize speedup.
- **Automated Testing:** Build a suite of tests with various PNG file formats.
- **Comparison with Sequential Results:** Validate that the parallelized algorithm produces output identical in quality and compliance to the sequential approach.