

Trabalho prático 2 - Em Grupo (máximo 3 pessoas)

1 Objetivo

O objetivo deste trabalho é implementar uma versão paralela do **Jogo da Vida de Conway** para ambientes de memória compartilhada, utilizando Pthreads. A simulação ocorrerá em uma matriz 2D que representa um mundo onde células podem estar vivas ou mortas.

2 Descrição da Simulação

O Jogo da Vida é um autômato celular com um conjunto simples de regras que governam a evolução de um sistema ao longo de gerações. O "mundo" é uma grade bidimensional de células, onde cada célula pode estar em um de dois estados: viva ou morta.

A simulação evolui em passos de tempo discretos (gerações). O estado de uma célula na próxima geração é determinado pelo estado de suas oito vizinhas (horizontal, vertical e diagonal) na geração atual, de acordo com as seguintes regras:

- **Subpopulação:** Uma célula viva com menos de duas vizinhas vivas morre.
- **Sobrevivência:** Uma célula viva com duas ou três vizinhas vivas sobrevive para a próxima geração.
- **Superpopulação:** Uma célula viva com mais de três vizinhas vivas morre.
- **Nascimento:** Uma célula morta com exatamente três vizinhas vivas torna-se viva.

3 Implementação

Primeiro, implemente uma versão sequencial do programa. Em seguida, desenvolva uma versão paralela utilizando Pthreads. A estratégia de paralelização deve se basear na **decomposição de domínio**, onde a matriz do mundo é dividida em seções (por exemplo, faixas de linhas ou colunas), e cada thread (Worker) é responsável por atualizar o estado das células em sua seção.

Um desafio central é o tratamento das **células de fronteira** entre as seções de diferentes Workers. A atualização de uma célula na borda de uma seção depende do estado das células vizinhas, que podem pertencer a outra seção. Isso exigirá um mecanismo de sincronização (barreiras) entre os Workers a cada geração para garantir que todos leiam o estado da geração anterior antes de começarem a escrever o novo estado.

3.1 Entrada e Saída

A aplicação deve ler de um arquivo de entrada os parâmetros da simulação:

- **N_GEN**: número de gerações para a simulação.
- **L** e **C**: número de linhas e colunas da matriz.
- **N**: número de células vivas iniciais.
- Seguido por **N** linhas, cada uma com as coordenadas **X Y** de uma célula viva.

A aplicação deve escrever na saída padrão o estado final do mundo após **N_GEN** gerações, no mesmo formato do arquivo de entrada.

4 Etapas e Relatório Técnico

A implementação e documentação do trabalho devem seguir 4 partes, que devem compor um relatório técnico:

1. **Perfil de desempenho sequencial**: Análise da versão sequencial usando ferramentas como `gprof`.
2. **Identificação das oportunidades de paralelização**: Justifique a escolha da estratégia de decomposição de domínio e discuta os desafios de sincronização.
3. **Paralelização**: Documente a implementação paralela, detalhando o uso de diretivas OpenMP ou funções Pthreads e as estruturas de dados compartilhadas.
4. **Avaliação dos ganhos da paralelização**: Apresente uma análise de desempenho da versão paralela, avaliando speedup e eficiência com diferentes tamanhos de matriz e número de threads.

5 Entrega

Os trabalhos devem ser entregues até a data marcada no portal didático. Devem ser entregues um documento PDF com o relatório e um arquivo compactado com o código-fonte em C, compilável com `gcc` (e a flag `-pthread`).