

Relatório do TP1

Aluno: Renan Antunes Braga Bomtempo

<https://github.com/RenanBomtempo/moving-threads>

Estruturas de Dados

A estrutura de dados é composta por 6 structs

1. `cell_t`

Estrutura da célula, que armazena todos os dados que podem estar contidos em cada posição do tabuleiro (id do grupo que ocupa a célula, contador do número de agentes na célula, mutex, variável de condição `empty` e variável de condição `occupied`, etc.).

2. `grid_t`

Armazena a dimensão N e tamanho total N^2 do tabuleiro, que é representado por um array unidimensional de `cell_t`.

3. `agent_t`

Contém o id do agente, id do grupo, a thread, e um array de `step_t`.

4. `step_t`

Cada passo da iteração passado no arquivo de entrada, contendo o tempo que a thread deve ficar inativa e a posição que deve ocupar.

5. `supervisor_t`

Estrutura que armazena todas os agentes em um array e o número de agentes.

Implementação

Processamento do input

O primeiro passo do programa é ler o arquivo de entrada e armazenar as informações nas estruturas de dado.

Inicialização do tabuleiro

A partir dos dados carregados, inicializamos as células todas vazias.

Inicialização dos agentes

Com o tabuleiro inicializado, agora criamos as threads de cada agente e posicionamos todos eles no tabuleiro em sua posição inicial. A partir de então, cada agente passa a executar seu trajeto.

Execução de cada agente

1. Cada agente espera o tempo necessário na posição inicial.
2. Para cada passo do trajeto:
 - a. O agente **entra** na próxima célula:
 - Dá lock na mutex da célula.
 - Incrementa o contador da célula.
 - Checa se existe outro agente do seu grupo ou se está vazia:
 - Caso esteja vazia, insere o id do grupo na célula e sinaliza com broadcast que a célula está ocupada.
 - Caso exista e seja do mesmo grupo, não faz nada.
 - Caso existe e não seja do mesmo grupo, espera até que a célula fique vazia.
 - Libera a mutex
 - b. O agente **sai** da célula que estava anteriormente:
 - Dá lock na mutex da célula.
 - Decrementa o contador da célula.
 - Checa se existe outro agente ou se está vazia:
 - Caso esteja vazia, remove o id do grupo na célula e sinaliza com broadcast que a célula está vazia.
 - Caso ainda exista outro agente na célula, não faz nada.
 - Libera a mutex
 - c. Aguarda o tempo definido no passo.

Término de execução

O programa termina assim que todos os agentes chegam ao final do trajeto e aguardam o ultimo tempo definido no passo.