

Desafio de Sistemas Operacionais: Concorrência

Objetivo

O objetivo deste desafio é criar um programa em python que consiga simular o gerenciamento de threads que concorrem por algum recurso compartilhado. O programa deve implementar um algoritmo de sincronização que evite condições de corrida e deadlock.

Requisitos

- O programa deve receber como entrada um número N de threads e um número M de recursos.
- Cada thread deve executar uma função que tente acessar um recurso aleatório por um tempo aleatório, e depois libere o recurso.
- O programa deve garantir que cada recurso só possa ser acessado por uma thread por vez, e que nenhuma thread fique esperando indefinidamente por um recurso.
- O programa deve imprimir na tela as ações de cada thread, como solicitação, alocação e liberação de recursos.
- O programa deve usar a biblioteca threading do python para criar e gerenciar as threads.
- O programa deve usar a classe Lock ou Semaphore da biblioteca threading para implementar o algoritmo de sincronização.

Avaliação

O programa será avaliado de acordo com os seguintes critérios:

- Funcionalidade: o programa deve executar corretamente e atender aos requisitos.
- Legibilidade: o código deve estar bem estruturado, comentado e seguir as boas práticas de programação.
- Eficiência: o programa deve evitar desperdício de recursos e tempo de execução.
- Originalidade: o programa deve ser de autoria própria e não copiado de outras fontes.
- O trabalho pode ser realizado por 2 alunos
- Entrega: 16/04/2024 até às 22:40
- Apresentar e postar o código no AVA na Semana 11