

Ficha Exercício 5: Pesquisa Concorrente de Docentes nos Ficheiros

Contexto

A Secretaria da Universidade de Mindelo (UM) enfrenta dificuldades no processo de pesquisa do nome de docentes em seus ficheiros de registro, pois a pesquisa atual é feita sequencialmente, um ficheiro por vez. Esse processo tem sido demorado, especialmente quando o nome procurado aparece em ficheiros mais ao final da lista.

Durante a aula de Programação Concorrente, os alunos do terceiro ano do curso de Engenharia Informática e Sistemas Computacionais (EISC) foram desafiados a desenvolver uma solução concorrente para esse problema.

Objetivo

Criar um programa que utiliza o padrão **Produtor-Consumidor** com **wait**, **notify** e **notifyAll**, para acelerar a busca de nomes de docentes em vários ficheiros ao mesmo tempo.

Cenário

1. Entrada:

- O nome do docente a ser pesquisado.
- Um conjunto de 9 ficheiros, cada um contendo 1000 nomes de docentes

2. Requisitos:

- **Produtor:** Um ou mais threads que leem os ficheiros e enviam os nomes para um buffer compartilhado.
- **Consumidor:** Um ou mais threads que retiram os nomes do buffer e verificam se coincidem com o nome procurado.
- Assim que o nome for encontrado, o programa deve:
 - Interromper todos os produtores e consumidores.
 - Retornar o nome do ficheiro e a linha onde o nome foi encontrado.

3. Saída

- Caso o nome seja encontrado, exibir:
 - Nome encontrado no ficheiro: <nome_do_ficheiro>, na linha: <número_da_linha>
- Caso o nome não seja encontrado, exibir:

- *Nome não encontrado em nenhum dos ficheiros.*

4. Regras para Implementação

- **Uso de threads:**
 - Cada thread produtor deve ler linhas dos ficheiros e adicionar ao buffer.
 - Cada thread consumidor deve retirar do buffer e verificar a coincidência com o nome procurado.
- **Sincronização:**
 - Utilize um buffer compartilhado protegido por métodos sincronizados, com wait, notify, e notifyAll para gerenciar a comunicação entre produtores e consumidores.
- **Término da Execução:**
 - Ao encontrar o nome, os consumidores e produtores devem ser notificados para interromper suas tarefas.
- **Tratamento de erros:**
 - Certifique-se de tratar exceções como interrupções de threads e manipulação de ficheiros inexistentes ou corrompidos.