

Trabalho prático - Escalonamento por Prioridades

Universidade Federal de Alfenas

Professor: Fellipe Guilherme

Discentes: Maria Eduarda Souza Nascimento 2023.1.08.029

Talita Oliveira Bielecky 2023.1.08.030

Documento Explicativo: Implementação de um Escalonador de Processos em Java

Introdução

Este documento descreve a implementação de um escalonador de processos em Java, utilizando biblioteca Swing para produzir uma interface gráfica interativa. O Escalonador é capaz de executar quatro algoritmos de escalonamento: First-Come, First-Served(FCFS), Round-Robin, Shortest Job First (SJF) e escalonamento por prioridade.

Estrutura do Código

Implementação dividida em três classes principais:

- **Classe Processos:** Um processo com atributos como:
 - ID
 - Tempo de execução
 - Tempo restante
 - Prioridade e Quantum.
- Métodos importantes
 - **executar(int tempo):** simula a execução do processo
 - **isFinalizado():** verifica se o processo foi concluído
- **Classe Escalonador:** Gerencia a fila de processos e contém métodos para cada algoritmo de escalonamento. Esses métodos implementam a lógica de cada algoritmo, com estruturas de dados como (**Queue**) e (**PriorityQueue**) para gerenciar os processos.

- **Classe Main:** Responsável pela interface gráfica. Utiliza `JFrame` e `JTextArea` para exibir a saída dos processos e permite a entrada de dados do usuário através de campos de texto e botões. A interação do usuário é capturada por meio de `ActionListener`

-

Bibliotecas

Essas bibliotecas foram usadas para que em conjunto permitissem a criação de uma interface gráfica que simula o escalonamento de processos. `Javax.swing` fornece a interface, `java.awt` organiza os componentes, `java.awt.event` é responsável pela interação do usuário e `java.util` permite gerenciar dados de forma eficiente

Implementação e Funcionalidade

Os processos são definidos manualmente pelo usuário através da interface gráfica. O usuário terá os campos de ID, tempo de execução, prioridades e quantum antes de cada processo para poder adicionar os números de acordo com seus parâmetros. Os algoritmos de escalonamentos são acionados pelos botões na interface, ao clicar no botão correspondente a um algoritmo, a execução é iniciada e os resultados são exibidos em um campo.

Uso de Estruturas de Dados

A escolha de `LinkedList` para FCFS e `PriorityQueue` para SJF e escalonamento por prioridade se deve à eficiência na manipulação de processos e à necessidade de manter a ordem de execução correta.

Instruções de Uso

Para executar o programa, será necessário ter o pacote Java Development Kit (JDK) instalado no pc, nosso código pode ser compilado e executados em qualquer ambiente que suporte java

Comandos:

`javac Main.java`

Executar

`java Main`

Conclusão

Este documento tem a finalidade de deixar mais claro o que pensamos durante a produção do nosso trabalho e também exemplificar como compilamos e interagimos na interface do nosso código.