

Relatório: Sincronização de Tarefas em Fases com Phaser

1. Objetivo

Este exemplo demonstra o uso de **Phaser** para coordenar múltiplas tarefas em fases. No cenário apresentado, diferentes threads realizam uma pesquisa de arquivos em diretórios específicos, filtram os resultados e, ao final, sincronizam suas fases antes de concluir o processamento.

2. Estrutura do Código

- **PesquisaArquivos:**
Responsável por pesquisar arquivos em um diretório cujo nome termina com uma extensão específica. A classe percorre recursivamente os diretórios, processa os arquivos, filtra os resultados com base na data de modificação e exibe as informações. Utiliza um objeto **Phaser** para sincronizar as fases de execução.
- **Principal:**
Configura o ambiente criando um **Phaser** com um número de participantes, instancia diferentes pesquisas de arquivos em diretórios distintos e inicia as threads correspondentes. Ao final, verifica se o **Phaser** foi encerrado.

3. Fluxo de Execução

1. Cada thread de **PesquisaArquivos** aguarda o início da fase utilizando `arriveAndAwaitAdvance()`.
2. As threads processam os diretórios e arquivos a partir do caminho inicial fornecido.
3. Cada thread verifica se obteve resultados:
 - Se não houver resultados, desregistra do **Phaser** e encerra.
 - Caso haja resultados, sincroniza com as demais threads e passa para a próxima fase.
4. As threads filtram os resultados com base na data de modificação e, novamente, sincronizam utilizando o **Phaser**.
5. Por fim, as threads exibem os resultados e se sincronizam antes de desregistrar do **Phaser**.

4. Exemplo de Execução

Saída do Console:

```
Sistema: Iniciando a pesquisa.  
Aplicativos: Iniciando a pesquisa.  
Documentos: Iniciando a pesquisa.  
Sistema: Fase 0: 5 resultados.
```

```
Aplicativos: Fase 0: 3 resultados.  
Documentos: Fase 0: 4 resultados.  
Sistema: Fase 1: 4 resultados.  
Aplicativos: Fase 1: 3 resultados.  
Documentos: Fase 1: 3 resultados.  
Sistema: C: \Windows\logfile.log  
Aplicativos: C: \Program Files\logfile.log  
Documentos: C: \Documents And Settings\logfile.log  
Encerrado: true
```

5. Conclusão

O uso do Phaser permite que múltiplas threads se sincronizem em pontos de controle definidos, garantindo que todas completem uma fase antes de prosseguir para a próxima. Essa abordagem é especialmente útil em cenários onde tarefas concorrentes precisam ser divididas em etapas, permitindo um controle refinado sobre a execução e facilitando o agrupamento e processamento dos resultados.