

---

# Videoconferência com Sincronização - Controle de Acesso Concorrente a Múltiplas Cópias de um Recurso



STATUS

DESENVOLVIDA

## Objetivo do Projeto



Este código simula uma videoconferência em que vários participantes chegam em momentos diferentes, e a conferência só começa quando todos os participantes estão prontos. O código utiliza dois mecanismos de sincronização: `CountDownLatch` e `CyclicBarrier`. O `CountDownLatch` garante que todos os participantes tenham chegado antes de iniciar a conferência, enquanto o `CyclicBarrier` garante que as threads dos participantes sejam sincronizadas antes de prosseguir para a etapa de início da conferência.

### Componentes do Sistema

1. Classe `Videoconference` Responsável por gerenciar a videoconferência e aguardar que todos os participantes cheguem. Utiliza o `CountDownLatch` para esperar que todos os participantes cheguem antes de iniciar a conferência.
2. Classe `Participant` Cada participante é representado por uma thread que simula sua chegada à conferência. O participante espera até que todos estejam prontos usando o `CyclicBarrier`, e então imprime sua preparação para iniciar a conferência.
3. Sincronização `CountDownLatch`: Usado na classe `Videoconference` para aguardar que todos os participantes cheguem antes de iniciar a conferência. O contador do `CountDownLatch` é decrementado a cada participante que chega. `CyclicBarrier`: Usado na classe `Participant` para garantir que todos os participantes alcancem o mesmo ponto antes de prosseguir, garantindo que todos estejam sincronizados para o início da conferência.



## Cenários teste



**Cenário 1: Execução Básica** Descrição: Execução do código com 10 participantes. Resultado Esperado: Todos os participantes chegam, a barreira é alcançada, e a conferência começa.

=====

Cenário 2: Atraso na Chegada dos Participantes Descrição: Um participante chega com um atraso significativo. Resultado Esperado: A conferência só começa após todos os participantes chegarem, incluindo o atrasado.

=====

Cenário 3: Menos Participantes que o Esperado Descrição: Menos participantes são criados do que o número esperado. Resultado Esperado: A conferência nunca começa, pois o CountdownLatch não chega a zero.

## Conclusão:



O código demonstrou ser eficaz na sincronização de threads utilizando CountdownLatch e CyclicBarrier. Os testes realizados garantiram que os mecanismos de sincronização funcionam conforme o esperado, evitando condições de corrida e assegurando que a conferência só começa quando todos os participantes estão prontos.

Considerações: CountdownLatch é adequado para cenários em que a conferência depende de todos os participantes estarem presentes. CyclicBarrier é útil para sincronizar múltiplas threads, garantindo que todas alcancem um ponto comum antes de seguir para a próxima etapa.

## Desenvolvedor



Marcio Fonseca



=====