

**INSTITUTO FEDERAL**

Paraíba

Campus Esperança

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA  
PARAÍBA – CAMPUS *ESPERANÇA*  
SISTEMAS OPERACIONAIS  
ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**Jorge Allan da Silva Santos  
Jéfter Lucas Lima da Silva  
Maria Eduarda Coelho da Silva**

**ESPERANÇA- PB**

**2024**

## **Leitores/Escritores - Sistemas Operacionais**

### **1. Resumo**

O problema de leitores e escritores envolve múltiplos processos competindo pelo acesso a um recurso compartilhado, onde leitores apenas leem e escritores alteram os dados. Este trabalho implementou uma solução utilizando semáforos para controlar o acesso simultâneo de leitores dos escritores. A abordagem é essencial em sistemas com dados compartilhados, como bancos de dados, e inclui a análise do desempenho.

### **2. Introdução**

Em sistemas operacionais, a sincronização de processos é um tema crucial para evitar problemas como condições de corrida, onde múltiplos processos competem pelo acesso aos mesmos recursos. Um caso específico de concorrência ocorre quando processos precisam acessar e modificar um recurso compartilhado, como um banco de dados ou uma tabela em memória. O problema de leitores e escritores é um exemplo clássico de como gerenciar esses acessos concorrentes.

### **3. O Problema dos Leitores e Escritores**

O problema dos leitores e escritores envolve dois tipos de processos:

- **Leitores:** Processos que apenas leem os dados compartilhados. Eles não alteram o conteúdo e, portanto, podem operar de forma concorrente, desde que não haja um processo escritor em execução.
- **Escritores:** Processos que alteram os dados compartilhados. Somente um escritor pode acessar os dados de cada vez, e um escritor não pode operar enquanto houver leitores ativos.

O objetivo é garantir que a leitura e a escrita ocorram de forma a evitar conflitos e garantir a integridade dos dados, promovendo a concorrência de forma eficiente.

### **4. Regras de Sincronização**

Para resolver o problema de leitores e escritores, devem-se seguir algumas regras básicas de sincronização:

- Leitura concorrente: Vários leitores podem ler os dados ao mesmo tempo, desde que nenhum escritor esteja ativo.
- Exclusividade na escrita: Somente um escritor pode ter acesso aos dados de cada vez, e não pode haver leitores enquanto um escritor estiver ativo.
- Prioridade: Pode-se implementar a solução de forma a dar prioridade aos leitores ou escritores, dependendo das necessidades do sistema. Por exemplo, se a prioridade for para os leitores, a leitura será mais eficiente, mas pode ocorrer uma negligência com os escritores.

## 5. Soluções para o Problema

Existem várias abordagens para resolver o problema de leitores e escritores, que podem ser implementadas com diferentes mecanismos de sincronização, como semáforos, monitores, e mutexes. No nosso programa feito em JavaScript, utilizamos a **Solução dos Semáforos**, que consistem em:

### 5.1. Solução com Semáforos

- Semáforos binários e contadores: Um semáforo binário pode ser usado para garantir que apenas um escritor tenha acesso exclusivo aos dados. Já um semáforo contador pode controlar quantos leitores estão acessando os dados de forma simultânea.
- Algoritmo básico: Um semáforo para o número de leitores, um para a exclusão de escritores e outro para controlar a quantidade de leitores ativos. Um leitor incrementa o contador de leitores e decrementa ao terminar a leitura, enquanto um escritor adquire a exclusividade antes de acessar os dados.

## 6. Metodologia Adotada

A implementação do problema de leitores e escritores foi realizada utilizando a linguagem de programação JavaScript, com o auxílio de semáforos binários e contadores para gerenciar o acesso ao recurso compartilhado. A metodologia adotada envolveu os seguintes passos:

1. Modelagem dos Processos: Foram definidos dois tipos de processos: leitores e escritores. Os leitores podem acessar os dados simultaneamente, desde que não haja

nenhum escritor em execução. Os escritores, por sua vez, têm acesso exclusivo aos dados e não podem operar enquanto houver leitores.

2. Implementação dos Semáforos: Utilizamos semáforos binários e contadores para controlar o número de leitores ativos e garantir a exclusividade do acesso para escritores. Cada leitor incrementa o contador de leitores ao acessar o recurso e o decrementa ao terminar a leitura. O escritor, antes de acessar os dados, adquire a exclusividade do recurso, bloqueando o acesso dos leitores.

3. Controle de Starvation: A solução inclui uma lógica para evitar starvation de escritores e leitores, priorizando o acesso de quem chegar primeiro seguindo o FIFO(primeiro a entrar, primeiro a sair) e realizando a exclusão mútua, ou seja, um buffer circular.

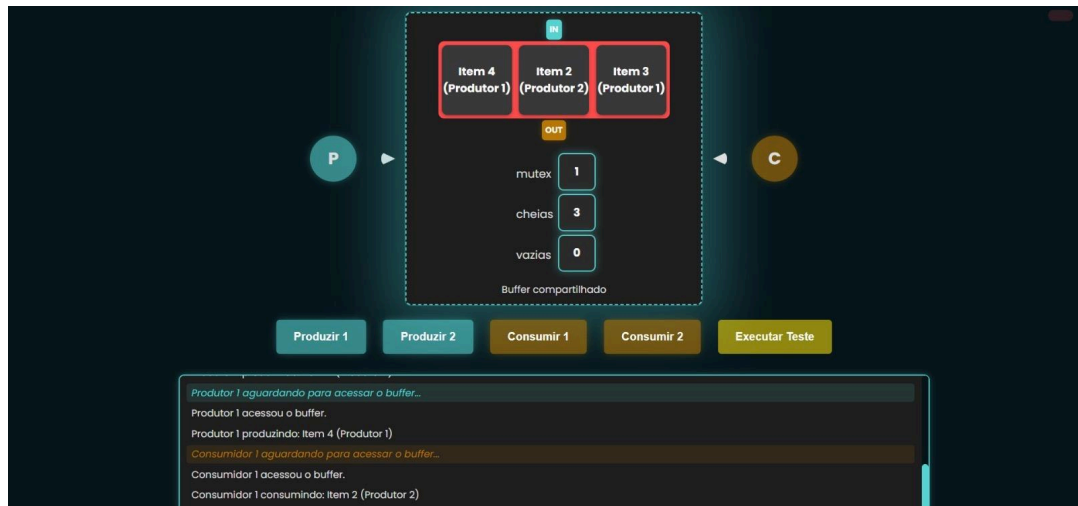
4. Simulação: O sistema foi simulado com múltiplos leitores e escritores concorrentes, com o objetivo de testar o comportamento do sistema sob condições de carga.

## **7. Aplicações Práticas**

O problema de leitores e escritores é fundamental em sistemas que manipulam bases de dados, sistemas de arquivos e caches. Em sistemas de gerenciamento de banco de dados, por exemplo, leitores podem realizar consultas simultâneas, enquanto atualizações são feitas por escritores. Essa sincronização adequada é essencial para manter a integridade dos dados e o desempenho do sistema.

## **8. Conclusão**

A solução para o problema de leitores e escritores é essencial para o desenvolvimento de sistemas operacionais e aplicações que necessitam de acesso concorrente a recursos compartilhados, dessa forma, optamos por escolher tal temática para nosso trabalho. A solução proposta para o problema de leitores e escritores, utilizando semáforos, demonstrou ser eficaz na sincronização de processos que acessam um recurso compartilhado. Segue abaixo a imagem da tela do nosso programa:



**Imagem 1:** Tela do programa Leitores/Escritores

## 8. Referências

- Silberschatz, A., Galvin, P. B., & Gagne, G. (2018). *Operating System Concepts*.
- Tanenbaum, A. S., & Bos, H. (2015). *Modern Operating Systems*.