

Sistemas Operacionais

Atividade Prática sobre
Comunicação entre
Processos

Prof. Rafael Oliveira Vasconcelos

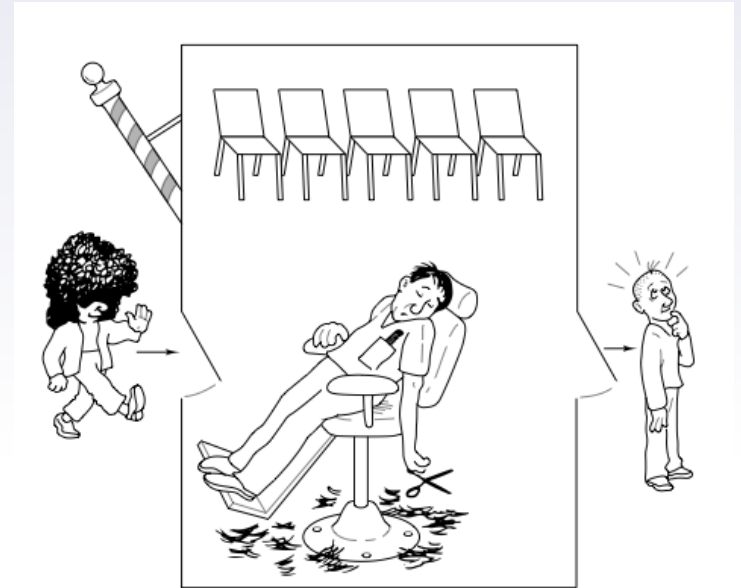
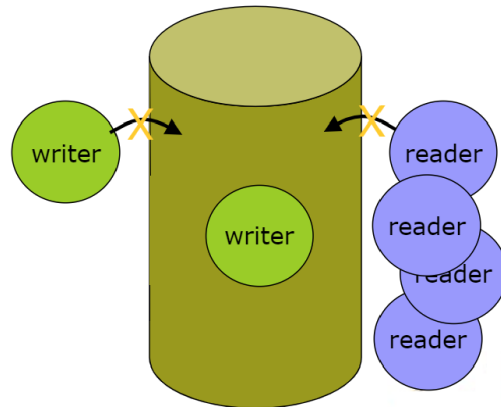
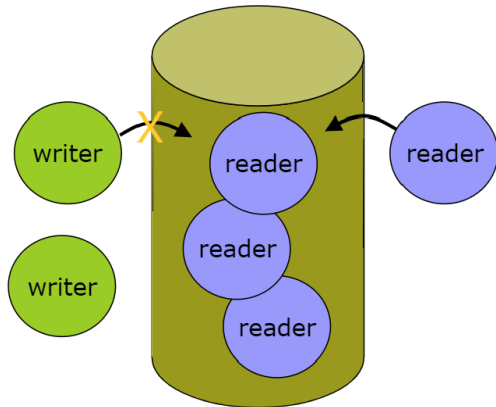
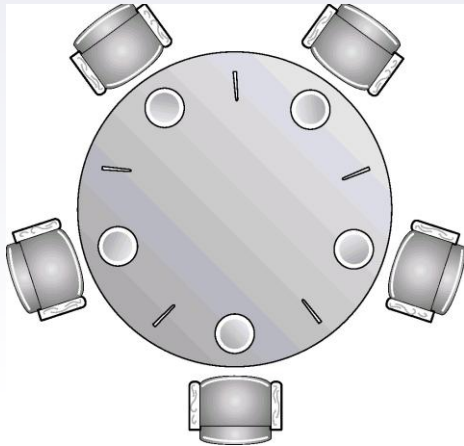


Relembrando a aula passada

Problemas Clássicos de
Comunicação entre Processos



Problemas Clássicos



1

Aula de Hoje

Atividade Prática sobre Comunicação
entre Processos



Informações

- ▶ Na próxima aula o grupo deve mostrar os programas funcionando e fazer uma breve explicação.
- ▶ A linguagem de programação é de livre escolha do grupo, entretanto, a exercício 3 cita classes do Java. Caso o grupo escolha uma linguagem de programação diferente, será preciso procurar uma classe com as mesmas características da classe Vector do Java ou implementar uma classe com tais características para que a comparação seja feita.

Exercício 1

- ▶ Implemente o problema do Barbeiro Dorminhoco
- ▶ Imagine que
 - ▶ Existem 2 barbeiros
 - ▶ A fila pode ter no máximo 10 clientes esperando
 - ▶ Caso o cliente chegue e a fila esteja cheia, exiba uma mensagem informativa
- ▶ Modele o seu programa para que
 - ▶ O corte de cabelo de um cliente demore um tempo aleatório entre 5s e 10s
 - ▶ Um novo cliente chegue aleatoriamente entre 4s e 6s

Exercício 2

- ▶ O seu desafio é implementar um `ArrayList` que seja `thread safe`. Lembre-se que as operações de consulta não causam condição de corrida umas com as outras, entretanto, as inserções e remoções causam condição de corrida entre elas.

Exercício 3

- ▶ Utilizando como base sua implementação **thread safe** do **ArrayList**, compare o desempenho com a versão original que **não é thread safe** utilizando apenas 1 thread
- ▶ Faça a comparação para os métodos de **inserção**, **busca** e **remoção**, variando o tamanho da lista e mostrando o tempo necessário para a realizar a operação com os tamanhos variados da lista. Adicionalmente, informe quantas operações (inserção, busca e remoção separadamente) podem ser realizadas por segundo em ambas as listas
- ▶ Repita os testes mas agora utilizando **16 threads** para comparar sua **implementação thread safe** com a **classe Vector**
- ▶ Cada thread realiza uma quantidade predefinida de operações de inserção, busca e remoção com valores aleatórios
- ▶ Informe os valores obtidos nos testes realizados

Exercício 4

- ▶ Na implementação de um Banco de Dados, há uma restrição para que no máximo 10 consultas sejam realizadas simultaneamente, ao passo que apenas 1 operação de escrita (insert, update ou delete) pode ocorrer simultaneamente.
- ▶ Caso uma 11ª consulta tente ser realizada, ela deve ser bloqueada até que alguma consulta finalize
- ▶ No momento da operação de escrita, não pode haver consultas no banco de dados
- ▶ Implemente uma classe que discipline o acesso ao Banco de dados
 - ▶ Implemente as 4 operações CRUD (Create, Read, Update e Delete)
- ▶ Crie um programa para testar e mostrar o funcionamento da(s) sua(s) classe(s)