



**UNIVERSIDADE FEDERAL DO CEARÁ  
CAMPUS QUIXADÁ**

Antônio Fernando de Araújo Costa - 494422

Ewaldo de Carvalho Bezerra Junior - 493521

Isânio Vitor Félix Araújo - 495231

Ítalo Rodrigues Nascimento - 494845

Odimar Sousa Falcão Filho - 496581

**SISTEMAS OPERACIONAIS  
Atividade prática sobre Comunicação entre Processos**

## SUMÁRIO

<b>Introdução</b>	<b>3</b>
<b>Desenvolvimento</b>	<b>3</b>
<b>Conclusão</b>	<b>3</b>

## 1. Introdução

Comunicação de processos são um grupo de mecanismos que permite aos processos transferir informação entre si. A execução de um processo pressupõe por parte do sistema operativo, entre outras coisas, a criação de um contexto de execução próprio que, de certa forma, abstrai o processo dos componentes reais do sistema. A comunicação é compreendida como um processo de compreender, compartilhar mensagens enviadas e recebidas, sendo que a própria mensagem e o modo como se dá seu intercâmbio exercem influência no comportamento dos envolvidos.

Baseado nisso, escolhemos o problema “**Barbeiro Adormecido**”, que consiste na ideia de manter o barbeiro ocupado enquanto houver clientes na barbearia, e descansando/esperando quando não há ninguém para ser atendido (fazendo isso de uma maneira ordenada). Cada vez que um cliente chega à barbearia e vê que nenhuma cadeira está vazia, decide não entrar na barbearia e vai embora, ademais, quando um cliente que está sentado na cadeira esperando, vê que o barbeiro terminou de cortar um cabelo, ele tenta ser o próximo a ter o cabelo cortado.

## 2. Desenvolvimento

Código do problema no arquivo Barbeiro.java.

## 3. Conclusão

Portanto, após a conclusão desta atividade, foi possível entender o funcionamento das threads no código com suas instruções na prática sendo compiladas e como os processos se comunicam nas etapas de execução. Dessa forma, as principais dificuldades enfrentadas foram a abstração deste problema, afinal, tivemos que criar classes para as funções do barbeiro e dos clientes a fim de conseguir correlacionar elas e aplicar corretamente as implementações de Runnable e métodos das threads de execuções com as pausas atreladas às exceções da linguagem Java. No geral, com um trabalho de cooperação mútua dos integrantes da equipe, conseguimos realizar com êxito este problema.