

**Cap.6 – Sincronização de Processos**

1. O que você entende por Race Condition? De forma geral, o que pode ser feito para evitar que ela aconteça?

**R-** Race Condition é a inconsistência/falha num sistema ou processo em que o resultado do processo é inesperadamente dependente da sequência ou sincronia de outros eventos. Acontece principalmente quando vários processos acessam recurso compartilhado de forma descontrolada.

Pode-se usar então da Seção Crítica para que processos possam cooperar entre si quando forem acessar um recurso compartilhado.

1. No contexto de Sistemas Operacionais, o que é uma Seção Crítica? Que exigências devem ser implementadas para resolver o “Problema da Seção Crítica”? Cite brevemente estes requerimentos.

**R-** A Seção Crítica nada mais é que qualquer recurso computacional (geralmente uma parte de código), que precisa ter acesso controlado para se evitar a Race Condition.

Devem ser implementadas 3 exigências:

Exclusão Mútua: se um processo Pi está executando em uma CS, então nenhum outro processo pode estar executando nesta CS;

Progresso: se um processo não está executando na sua CS, então ele não pode impedir que outro processo entre nela;

Espera Limitada: todo processo deve possuir um tempo limite de espera. Se ele precisa usar uma CS, então dentro de um tempo ele tem que usar esta CS;

1. O que você entende por Espera em Ação? O que pode ser feito para evitá-la?

**R-** A Espera em Ação pode ser definida como: Enquanto um Processo estiver fazendo uso de uma Seção Crítica, qualquer outro que tentar entrar na mesma Seção deverá percorrer um looping até o semáforo ser liberado. Para evitá-la, uma alternativa poderia ser bloquear o processo (suspendê-lo enquanto espera).

1. O que são Semáforos? Explique sucintamente e indique quais os tipos de semáforos existentes.

**R-** Semáforos são ferramentas de sincronização, também de software, compostas por uma variável inteira, acessada somente (fora a inicialização) por duas operações atômicas: wait e signal.

Existem Semáforos Binários e Semáforos de Contagem.

1. Cite 3 problemas clássicos que podem ser resolvidos utilizando de semáforos e explique sucintamente como eles são resolvidos.

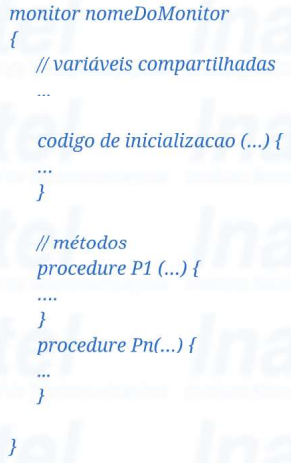
**R-** Problema do Buffer Limitado: Pode-se resolver o problema do buffer limitado de forma eficiente usando um mutex e dois semáforos, um para cada aspecto de coordenação envolvido. Esse problema também pode ser resolvido com variáveis de condição. Além do mutex para acesso exclusivo ao buffer, são necessárias variáveis de condição para indicar a presença de itens e de vagas no buffer.

Problema dos Leitores/Gravadores: Uma solução simplista para esse problema consistiria em proteger o acesso a área compartilhada com um mutex ou semáforo inicializado em 1; assim, somente um processo por vez poderia acessar a área, garantindo a integridade de todas as operações.

Problema do Jantar do Filósofos: Uma solução trivial para o problema do jantar dos filósofos consiste em colocar um “saleiro” hipotético sobre a mesa: quando um filósofo deseja comer, ele deve obter o saleiro antes de obter os palitos; assim que tiver ambos os palitos, ele devolve o saleiro à mesa e pode comer. (Obviamente, a solução do saleiro serializa o acesso aos palitos e por isso tem baixo desempenho se houverem muitos filósofos disputando o mesmo saleiro).

1. O que são Monitores? De forma geral, como eles podem ser implementados?

**R-** Monitores são uma espécie de Classe, ou seja, um conjunto de variáveis, métodos, e estruturas de dados, capaz de fornecer automaticamente ao programador o controle de uma Seção Crítica;



1. Que tipo de solução pode ser feita em Java para prover a exclusão mútua de métodos de um programa? Explique.

**R-** Pode ser feita, para exclusão mútua, a utilização de um semáforo binário, isto é, que só pode assumir dois valores distintos, 0 e 1. O travamento por semáforo deve ser feito antes de utilizar o recurso, e após o uso o recurso deve ser liberado. Enquanto o recurso estiver em uso, qualquer outro processo que o utilize deve esperar a liberação.