



PROCESSOS E THREADS

Sistemas Operacionais
Prof. Jeferson Silva
Engenharia de Software

GERENCIAMENTO DE PROCESSOS

Sistemas Operacionais gerenciam processos que executam simultaneamente

Comunicação Interprocesso, Pipeline e Condições de Corrida são conceitos fundamentais

COMUNICAÇÃO INTERPROCESSO (IPC)

Processos precisam se comunicar uns com os outros para cooperar

IPC refere-se a técnicas para trocar informações entre processos

IPC VIA MEMÓRIA COMPARTILHADA

Região de memória compartilhada é alocada pelo SO

Os processos podem ler e escrever nesta região de memória compartilhada

IPC VIA PIPES

Pipes são usados para IPC unidirecional (ou seja, um processo escreve e outro lê)

Pipes são criados usando a função `pipe()` e podem ser usados para comunicação entre processos relacionados

IPC VIA MENSAGENS

Mensagens são uma técnica de IPC onde os processos enviam e recebem mensagens

O SO gerencia as mensagens e garante que elas sejam entregues corretamente

IPC VIA SINAIS

Sinais são interrupções que um processo pode enviar a outro

Os sinais são usados para notificar outros processos de eventos importantes

IPC VIA SEMÁFOROS

Semáforos são usados para controlar o acesso a recursos compartilhados

O semáforo é um objeto que possui um valor inteiro que pode ser alterado pelos processos

SEMÁFOROS

São objetos que permitem que apenas um número específico de threads acessem uma seção crítica do código de cada vez.

Eles têm uma contagem associada, que é aumentada ou diminuída por threads que estão entrando ou saindo da seção crítica.

Quando a contagem atinge zero, todos os threads que estão tentando entrar na seção crítica são bloqueados até que outro thread saia.

IPC VIA SOCKET

Sockets são uma técnica de IPC para comunicação entre processos em diferentes máquinas

Os sockets podem ser usados para comunicação em rede e geralmente são implementados usando TCP/IP

PIPELINE

Pipeline é uma técnica de IPC em que um processo envia a saída para outro processo como entrada

Cada processo pode ser executado em um momento diferente e, portanto, o pipeline pode ser usado para processamento em lote

CONDIÇÕES DE CORRIDA

Condições de corrida ocorrem quando dois ou mais processos acessam o mesmo recurso compartilhado ao mesmo tempo

Isso pode levar a resultados inesperados e comportamentos não determinísticos

MUTEX

Um *Mutex* é um objeto que age como um portão para uma seção crítica do código.

Quando um thread quer acessar esta seção crítica, ele deve primeiro adquirir o *Mutex*.

Se outro thread já tiver adquirido o *Mutex*, então o thread atual é bloqueado e aguarda até que o *Mutex* esteja liberado.

Quando o *Mutex* é liberado pelo thread que o adquiriu, o próximo thread na fila pode adquiri-lo e assim por diante.

MONITORES

Um monitor é um objeto que envolve um conjunto de variáveis e procedimentos que são acessados exclusivamente por threads.

O monitor garante que apenas um thread possa acessar as variáveis e procedimentos de cada vez.

Quando um thread deseja acessar o monitor, ele deve primeiro adquirir o controle do monitor.

Se outro thread já tiver o controle, o thread atual é bloqueado até que o controle seja liberado.