

Programação Concorrente



Programação Concorrente

[O que é]

- “Um programa concorrente é um conjunto de programas seqüenciais comuns que são executados em um paralelismo abstrato”
(M.Bem-Ari)

Programação Concorrente

[O que é]

- “Um programa concorrente especifica 2 ou mais processos que cooperam para realizar uma tarefa. Processos cooperam através de comunicação; utilizam variáveis compartilhadas ou troca de mensagens“(G. R. Andrews)

Programação concorrente: introdução

- O mundo real funciona concorrentemente: várias atividades podem ser executadas em paralelo. Exemplo: uma pessoa pode estar
 - respirando, e,
 - falando, e
 - escrevendo, e
 - lendo, etc.
- Computadores também operam concorrentemente. Exemplo: um computador pode estar
 - compilando um programa, e
 - recebendo uma mensagem, e,
 - imprimindo um arquivo, e,
 - tocando música, etc.

Programação Concorrente

[Motivação]

- Aproveitar hardware com múltiplos processadores
- Atender a vários usuários simultaneamente
- Melhorar o desempenho das aplicações
- Aumentar a disponibilidade para o usuário
- Objetos ativos e controle de atividades
- Programas paralelos

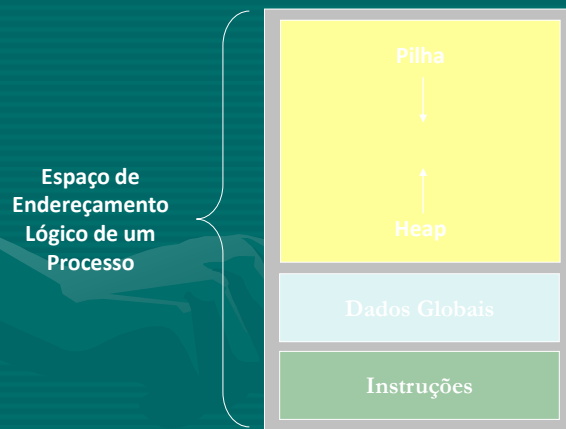
Programação Concorrente

[Processos]

- Um processo é um programa que está em algum estado de execução
- Tem espaço de endereçamento próprio, que é mapeado pelo S.O. para memória física
- Possui um fluxo de controle ou thread único
- Mantém um contador de programa (PC) que indica o endereço da próxima instrução
- A MMU (Memory Management Unit) traduz os endereços lógicos em endereços físicos, que normalmente não são contíguos (Memória Virtual)

Programação Concorrente [Processos]

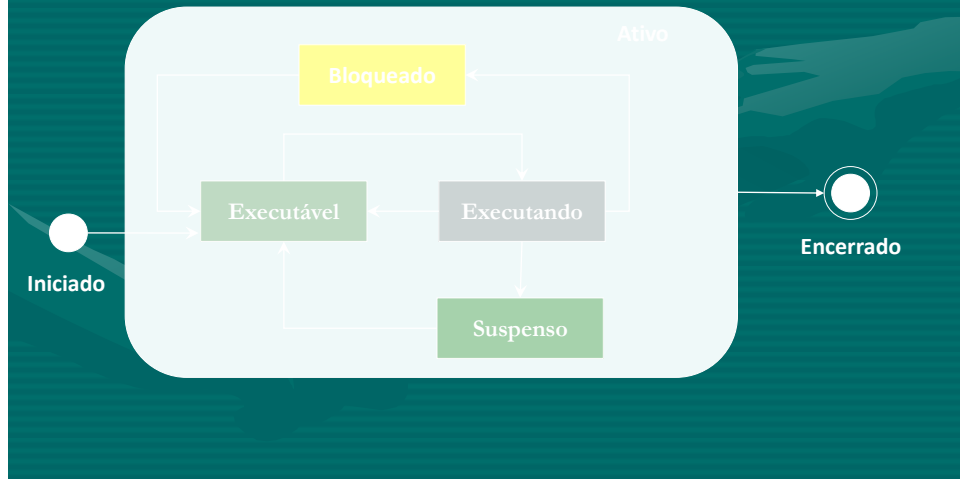
- Espaço de Endereçamento Lógico



Programação Concorrente [Processos]

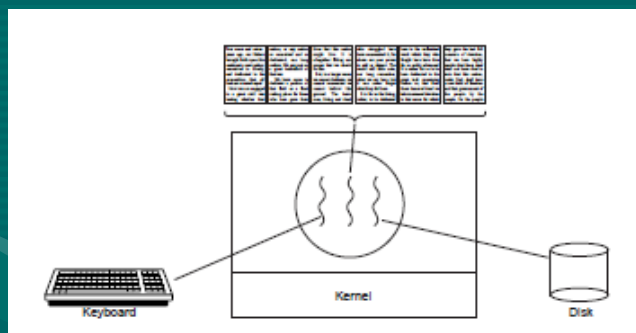
- Estados de um processo
 - **Executando** (Running): Utilizando a CPU
 - **Executável ou Pronto** (Runnable ou Ready): Esperando para ser escalonado para usar a CPU
 - **Suspensão** (Suspended): Recebeu um sinal para ser suspenso
 - **Bloqueado** (Blocked): Esperando pela conclusão de algum serviço solicitado ao S.O.

Programação Concorrente [Processos]



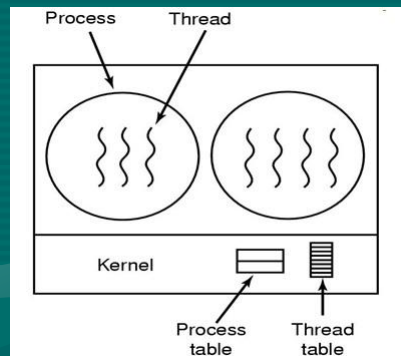
Programação Concorrente [Threads]

- O que é uma thread e como funciona?



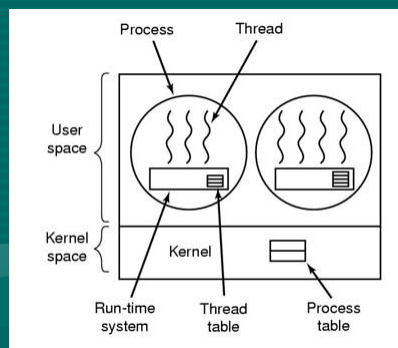
Programação Concorrente [Threads]

- *Como funciona o Kernel-Level Thread (KLT)?*



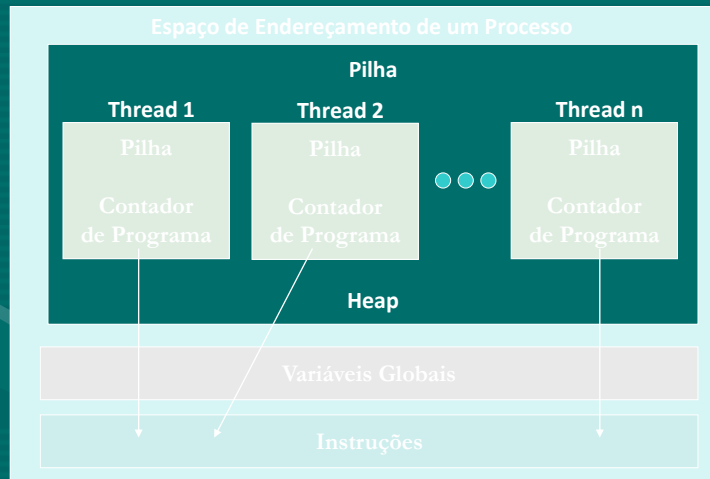
Programação Concorrente [Threads]

- *Como funciona o User-Level Thread (ULT)?*



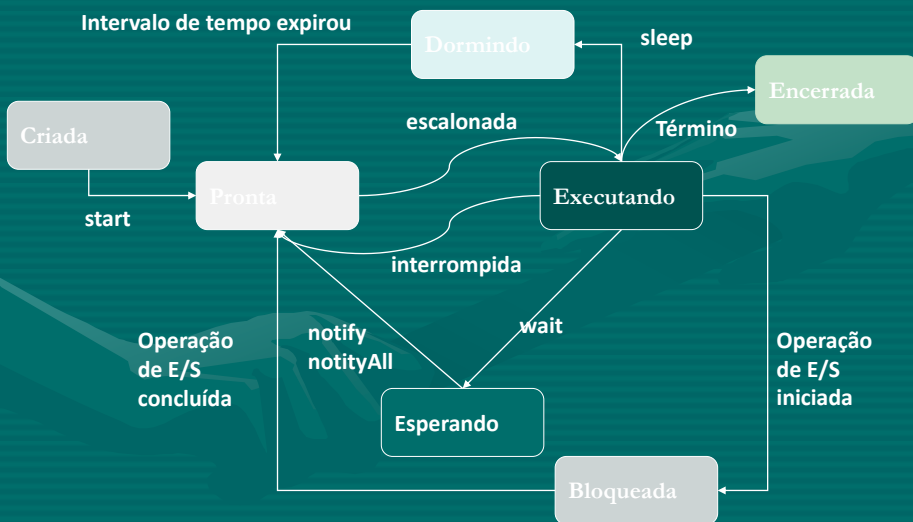
Programação Concorrente

[Threads]



Threads

[Ciclo de Vida]



Programação Concorrente

[Threads]

- Vantagens sobre o uso de Threads:
 - São muito mais leves de serem criadas
 - A troca de contexto é mais suave pois compartilha instruções, *heap* e variáveis globais
 - Facilitam o compartilhamento de memória