



---

---


---

---


---

---

---



### Plano de Ensino



- Introdução: Conceitos e Definições.
- Processos, Comunicação e Sincronização de Sistemas Distribuídos.
- **Processamento Paralelo e Multiprocessadores.**
- Pseudo-Paralelismo
- Tolerância a falhas.
- Aplicações Distribuídas: Socket.
- Objetos Distribuídos: RMI.

---

---


---

---


---

---

---



### Livro-Texto



- Bibliografia Básica:
  - » TANENBAUM, Andrew S; STEEN, Maarten Van. Sistemas Distribuídos : Princípios e Paradigmas. 2ª Ed. São Paulo: Pearson Prentice Hall, 2007.
- Bibliografia Complementar:
  - » DANTAS, Mario. Computação Distribuída de Alto Desempenho : redes, clusters e grids computacionais. 1ª Ed. Rio de Janeiro: Axxel Books, 2005.
  - » SILBERSCHATZ, Abraham; GALVIN, Peter Baer; GAGNE, Greg. Sistemas Operacionais - Conceitos e Aplicações. 1ª Ed. Rio de Janeiro: Campus, 2001.

---

---

---

---

---

---

---

### 3. Proc. Paralelo – Introdução



- É o uso simultâneo de um ou mais CPUs para executar um programa.
  - » O processamento paralelo faz com que um programa execute mais rapidamente porque utiliza de forma mais otimizada a CPU.
  - » Na prática é difícil dividir um programa de tal modo que a CPU possa executar diferentes partes sem interferência mútua.
- É possível conectar computadores com CPUs simples em rede.
  - » Necessita de software sofisticado para processamento distribuído.
- Processamento paralelo difere do conceito de *multitasking*, onde uma CPU simples executa vários programas simultaneamente → *pseudo-paralelismo*.

---

---

---

---

---

---

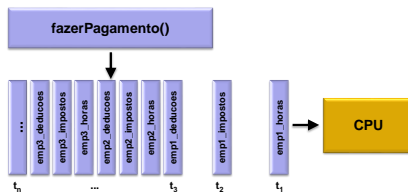
---

---

### 3. Proc. Paralelo – Introdução



- Tradicionalmente, softwares são escritos para processamento serial.
  - » Para rodar em um único computador com uma única CPU.
  - » Um problema é quebrado em uma série discreta de instruções.
  - » Instruções são executadas uma após a outra.
  - » Somente uma instrução pode executar em um dado momento.



---

---

---

---

---

---

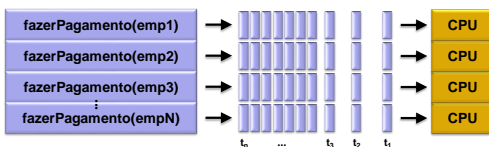
---

---

### 3. Proc. Paralelo – Introdução



- De um modo simples, o processamento paralelo é o uso simultâneo de múltiplos recursos para resolver um problema computacional:
  - » Um problema é quebrado em partes discretas que podem ser resolvidas concorrentemente.
  - » Cada parte pode ser também quebrada em uma série de instruções.
  - » As instruções de cada parte executam simultaneamente em diferentes CPUs.



---

---

---

---

---

---

---

---

### 3. Proc. Paralelo – Introdução



- Os recursos de um computador podem ser:
  - » Um computador simples com múltiplos processadores.
  - » Um número arbitrário de computadores conectados por rede.
  - » Uma combinação de ambos.
- O problema computacional deve ser capaz de:
  - » Ser quebrado em pedaços discretos de trabalho que podem ser resolvidos simultaneamente.
  - » Executar múltiplas instruções de programa a qualquer momento.
  - » Ser resolvidos em um tempo menor com múltiplos recursos computacionais do que com um simples recurso.

---

---

---

---

---

---

---

### 3. Proc. Paralelo – Introdução



- Processamento paralelo é uma evolução da computação serial para simular o que sempre ocorre no mundo natural: eventos interrelacionados e complexos que acontecem ao mesmo tempo, mesmo dentro de uma sequência. Por exemplo:
  - » Formação da Galáxia.
  - » Movimentos planetários.
  - » Padrões do tempo e oceanos.
  - » Movimentação das placas tectônicas.

---

---

---

---

---

---

---

### 3. Proc. Paralelo – Taxinomia de Flynn



- Existem diferentes formas de classificação do processamento paralelo; porém uma das mais utilizadas é a Taxinomia de Flynn, desde 1966.
- Distingue arquiteturas de computador com multi-processadores de acordo com 2 dimensões independentes: Instrução e Dados.
- Cada uma destas dimensões pode ter somente um dos dois estados: Simples e Múltiplo.

<b>SISD</b> Instrução Simples, Dado Simples	<b>SIMD</b> Instrução Simples, Dados Múltiplos
<b>MISD</b> Instruções Múltiplas, Dado Simples	<b>MIMD</b> Instruções Múltiplas, Dados Múltiplos

---

---

---

---

---

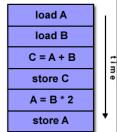
---

---

### 3. Proc. Paralelo – Taxinomia de Flynn



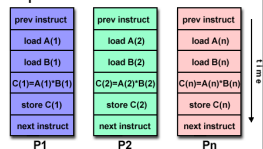
- **Instrução Simples, Dado Simples (SISD – Single Instruction, Single Data)** → um computador serial (não-paralelo).
  - » Instrução Simples: somente uma cadeia de instrução está sendo processado pela CPU durante qualquer ciclo de clock.
  - » Dado Simples: somente uma cadeia de dados está sendo utilizado como entrada durante qualquer ciclo de clock.
- Execução determinística.
- É o mais antigo e o mais comum tipo de computador ainda utilizado.
  - » Exemplos: antigas gerações de mainframes, minicomputadores e workstations, maioria dos modernos PCs atuais.



### 3. Proc. Paralelo – Taxinomia de Flynn



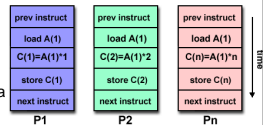
- **Instrução Simples, Dados Múltiplos (SIMD – Single Instruction, Multiple Data)** → um tipo de computador com processamento paralelo.
  - » Instrução Simples: toda a unidade de processamento executa a mesma instrução em qualquer ciclo de clock.
  - » Dados Múltiplos: cada unidade de processamento pode operar diferentes elementos de dados.
- Específico para resolver problemas característicos com algo grau de regularidade como processamento de imagens, execução síncrona e determinística.
  - » Exemplos: CM-2, ILLIAC IV, IBM 9000, Cray X-MP, computadores modernos particularmente com placas gráficas (GPUs).



### 3. Proc. Paralelo – Taxinomia de Flynn



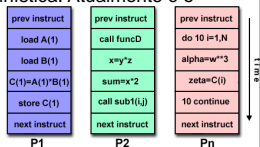
- **Instruções Múltiplas, Dado Simples (MISD – Multiple Instruction, Single Data)**: um tipo de computador paralelo.
  - » Instruções múltiplas: cada unidade de processamento opera sobre o dado através de cadeias separadas de instrução.
  - » Dado simples: uma cadeia simples de dados é carregada em múltiplas unidades de processamento.
- Esta classe de computador não existe atualmente.
  - » Exemplo: um computador experimental foi o Carnegie-Mellon C.mmp (1971).
- Utilizações possíveis:
  - » Múltiplos filtros de frequência operando em um único sinal.
  - » Vários algoritmos de criptografia tentando quebrar uma única mensagem codificada.



### 3. Proc. Paralelo – Taxinomia de Flynn



- **Instruções Múltiplas, Dados Múltiplos (MIMD – Multiple Instruction, Multiple Data)** → um tipo de computador paralelo.
  - » Instruções Múltiplas: cada processador pode estar executando uma cadeia de instruções diferente.
  - » Dados Múltiplos: cada processador pode estar trabalhando com diferentes cadeias de dados.
- Execução pode ser síncrona ou assíncrona, determinística ou não-determinística. Atualmente é o tipo de computador paralelo mais comum.
  - » Exemplos: a maioria dos supercomputadores modernos, clusters de rede e grids, computadores SMP e PCs multi-core.



### 3. Proc. Paralelo – Arquiteturas de Memória

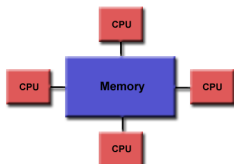


- **Memória Compartilhada** → possui grande variação mas possui em comum a habilidade de todos os processadores em acessar toda a memória como um único espaço de endereçamento global.
  - » Múltiplos processadores podem operar de forma independente mas dividem os mesmos recursos de memória.
  - » Alterações em uma posição de memória por um processador torna-se visível para todos os outros.
  - » Máquinas com memória compartilhada podem ser divididas em 2 classes principais baseado no tempo de acesso: UMA e NUMA.

### 3. Proc. Paralelo – Arquiteturas de Memória



- **UMA (Uniform Memory Access):**
  - » Representado atualmente pelas máquinas SMP (Symmetric Multiprocessor).
  - » Processadores idênticos.
  - » Acesso e tempos de acesso à memória iguais.

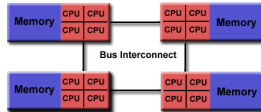


### 3. Proc. Paralelo – Arquiteturas de Memória



#### ▪ NUMA (Non-Uniform Memory Access):

- » Geralmente construído pela ligação de dois ou mais SMPs.
- » Um SMP pode acessar diretamente a memória de outro SMP.
- » Nem todos os processadores tem tempo de acesso igual para toda a memória.
- » Acesso à memória através de link é mais lenta.



---

---

---

---

---

---

---

---

### 3. Proc. Paralelo – Arquiteturas de Memória



#### ▪ Vantagens do compartilhamento de memória:

- » Espaço de endereçamento global fornece uma perspectiva de programação amigável para acesso à memória.
- » Compartilhamento de dados entre tarefas é rápida e uniforme devido à proximidade das memórias com a CPU.

#### ▪ Desvantagens do compartilhamento de memória:

- » Principal desvantagem é a falta de escalabilidade entre memória e CPU. Adicionar mais CPUs pode aumentar o tráfego geometricamente entre a memória e o processador.
- » Responsabilidade do programador na construção de sincronização que garanta o acesso "correto" à memória global.
- » Custo: torna-se cada vez mais difícil e caro projetar e produzir máquinas de memória compartilhada com o crescente número de processadores.

---

---

---

---

---

---

---

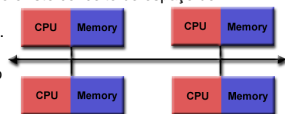
---

### 3. Proc. Paralelo – Arquiteturas de Memória



#### ▪ Memória Distribuída → sistemas de memória distribuídas necessitam de uma rede de comunicação para conectar a memória ao processador.

- » Processadores tem seu próprio local de memória. Os endereçamentos de memória de um processador não mapeiam o de outro processador. Não existe conceito de espaço de endereçamento global através dos processadores.
- » Devido ao fato do processador ter seu próprio local de memória ele opera independentemente. Alterações são realizadas na sua memória local sem efeito sobre os demais.
- » A troca e sincronização de informações entre os processadores é de responsabilidade do programador.



---

---

---

---

---

---

---

---

### 3. Proc. Paralelo – Arquiteturas de Memória



#### ▪ Vantagens da Memória Distribuída:

- » Memória é escalável com o número de processadores. Aumentando o número de processadores, aumenta o tamanho da memória proporcionalmente.
- » Cada processador pode rapidamente acessar sua própria memória sem interferência e sem sobrecarga.
- » Custo-benefício: pode usar processadores e redes sob demanda, terceirizando se necessário.

#### ▪ Desvantagens da Memória Distribuída:

- » O programador é responsável por muitos dos detalhes associados com a comunicação de dados entre os processadores.
- » Pode ser difícil mapear estruturas de dados existentes, baseado na memória global, para este tipo de organização de memória.
- » Tempos de acesso às memórias é não uniforme (NUMA).

---

---

---

---

---

---

---

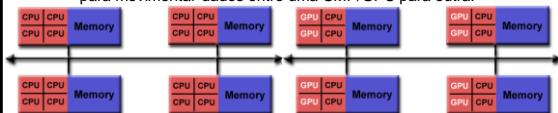
---

### 3. Proc. Paralelo – Arquiteturas de Memória



#### ▪ Memória Híbrida Compartilhada-Distribuída → os computadores mais rápidos atualmente empregam arquiteturas de memória compartilhadas e distribuídas.

- » O componente de memória compartilhada pode ser um SMP ou uma unidade de processamento gráfico (GPU).
- » O componente de memória distribuída é uma rede de múltiplas máquinas SMP/GPU, que conhecem apenas sua própria memória. Entretanto, comunicações de rede são necessárias para movimentar dados entre uma SMP/GPU para outra.



- » Vantagens e Desvantagens: o que já foi listado em comum a ambas as arquiteturas.

---

---

---

---

---

---

---

---



Sistemas Distribuídos – Aula 03

[clayton.valdo@anhanguera.com](mailto:clayton.valdo@anhanguera.com)



---

---

---

---

---

---

---

---