

INE 5410

## Programação Concorrente

Professor:

Lau Cheuk Lung (turma A)

INE – UFSC

lau.lung@inf.ufsc.br

## Conteúdo Programático

1. Introdução
2. Programação Concorrente
3. Sincronização
  1. Condição de corrida, região crítica e exclusão mútua
  2. Monitores, Locks e semáforos
  3. Troca de mensagens
  4. Justiça (como evitar *starvation*)
  5. deadlocks

## Conteúdo Programático

1. Problemas clássicos de programação concorrente
  1. Produtor/consumidor
  2. Jantar dos filósofos
  3. Barbeiro dorminhoco

## Recursos Computacionais

- Software  
J2SE SDK
- Página da disciplina <http://www.inf.ufsc.br/~lau.lung/INE5410>
- Lista de e-mails  
[ine5410-0538a@inf.ufsc.br](mailto:ine5410-0538a@inf.ufsc.br)

Unidade 1

## Introdução à Programação Concorrente

- Programação Paralela e Concorrente
- Vantagens e Dificuldades
- Plataformas de Execução
- Suporte Computacional

## Onde estamos no curso ?

Aplicações
Computação Distribuída
Prog. Concorrent
Sistemas Operacionais
Redes de Computadores

6

## Panorama Atual

Poder de processamento das máquinas vem crescendo rapidamente

Grande parte das máquinas interligada por redes de computadores

Programação Paralela e Distribuída

Sistemas e aplicações estão cada vez mais complexos

- Funcionalidade, Interfaceamento gráfico, Comunicação, ...
- Maior carga, Maior número de usuários, ...
- Melhor tempo de resposta, Maior confiabilidade ...

7

## Introdução

- Software é aplicado a um número cada vez maior de áreas da atividade humana;
- Exigência de qualidade (confiabilidade) e desempenho são fatores cruciais;
- Aplicações críticas, software para controle de:
  - Usina nuclear, termelétrica, geração e distribuição de energia;
  - Distribuição e controle da qualidade da água;
  - Tráfego aéreo, marítimo, sistema de transporte ferroviário, metrô;
  - Hospitalar, monitoramento de paciente;

8

## Introdução

- Em muitas áreas de aplicação, as características intrínsecas ao problema sendo tratado requer a utilização de mecanismos de programação que contemple aspectos como:
  - Paralelismo
  - Não determinismo
  - Comunicação
  - Sincronização

9

## Introdução

- O suporte ao paralelismo é necessário para aumentar o desempenho de um sistema e ao mesmo tempo permitir a implementação de soluções que só seriam possíveis a partir do paralelismo
- Níveis
  - Processador
  - Arquitetura do sistema computacional
  - Sistemas Operacionais
  - Infra-estrutura computacional
  - Linguagens de programação

10

## Programação Paralela

- O que é?
  - Consiste em executar **simultaneamente** várias partes de uma mesma aplicação;
  - Eventos (tarefas) possuem independência entre eles:
    - Não existe uma ordenação entre eles na operação global do sistema;
  - Tornou-se possível a partir do desenvolvimento de sistemas operacionais multi-tarefa, multi-thread e paralelos;
  - Um sistema de computação paralelo é um computador com mais de um processador para processamento paralelo.

11

## Programação Paralela

- Aplicações são executadas paralelamente:
  - Em um mesmo processador
  - Em uma máquina multiprocessada
  - Em um grupo de máquinas interligadas que se comporta como uma só máquina

12

## Programação Concorrente

- Execução concorrente está associada ao modelo de **um servidor** atendendo a **vários clientes** através de uma política de escalonamento no tempo.
- Em execução paralela é possível ter concorrência: modelo de vários servidores atendendo a vários clientes simultaneamente ao mesmo tempo.
  - Condições necessárias:
    - número de clientes > número de recursos
    - acesso simultâneo ao mesmo recurso
- As linguagens de programação podem então ser classificadas como sequenciais, paralelas e concorrentes.

13

## Diferenças

- Acoplamento
  - Sistemas paralelos são fortemente acoplados: compartilham hardware ou se comunicam através de um barramento de alta velocidade
  - Sistemas distribuídos são fracamente acoplados



14

## Diferenças

- Previsibilidade
  - O comportamento de sistemas paralelos é mais previsível; já os sistemas distribuídos são mais imprevisíveis devido ao uso da rede (AP, Hub, Switch, Roteadores, Firewall, etc.) e a falhas.



15

## Diferenças

- Influência do Tempo
  - Sistemas distribuídos são bastante influenciados pelo tempo de comunicação pela rede; em geral não há uma referência de tempo global
  - Em sistemas paralelos o tempo de troca de mensagens pode ser desconsiderado
- Controle
  - Em geral em sistemas paralelos se tem o controle de todos os recursos computacionais; já os sistemas distribuídos tendem a empregar também recursos de terceiros



16

## Vantagens

- Usam melhor o poder de processamento
- Apresentam um melhor desempenho
- Permitem compartilhar dados e recursos
- Podem apresentar maior confiabilidade
- Permitem reutilizar serviços já disponíveis
- Atendem um maior número de usuários
- ...



17

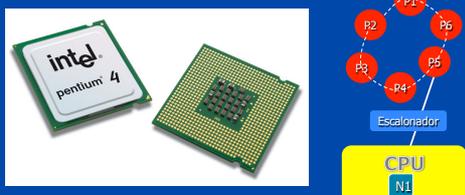
## Dificuldades

- Desenvolver, gerenciar e manter o sistema
- Controlar o acesso concorrente a dados e a recursos compartilhados
- Evitar que falhas de máquinas ou da rede comprometam o funcionamento do sistema
- Garantir a segurança do sistema e o sigilo dos dados trocados entre máquinas
- Lidar com a heterogeneidade do ambiente
- ...

18

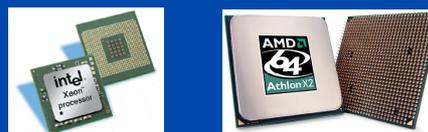
## Plataformas de Execução

- Um S.O. multitarefa permite simular o paralelismo em um único processador, alternando a execução de processos



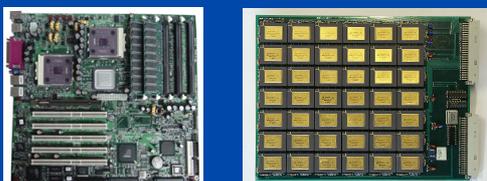
## Plataformas de Execução

- Um processador com núcleo múltiplo permite paralelismo real entre processos, executando múltiplas instruções por ciclo



## Plataformas de Execução

- Uma Placa-Mãe Multiprocessador permite que cada processador execute um processo



## Plataformas de Execução

- Um Cluster é uma solução de baixo custo para processamento de alto desempenho



## Suporte Computacional

- Suportes para Computação Paralela e Concorrente devem fornecer:
  - Mecanismos para execução paralela de programas
  - Mecanismos para controle de concorrência
  - Mecanismos para comunicação entre processos/threads paralelos
  - Ferramentas e mecanismos para desenvolvimento, testes, gerenciamento, controle, segurança, tolerância a faltas, etc.

23

## Suporte Computacional

- Suporte para Computação Paralela
  - Sistemas Operacionais Multi-Tarefa: permitem a troca de contexto entre processos / threads. Ex.: Windows, Linux, Solaris, HP-UX, AIX, etc.
  - Linguagens Multi-Tarefa: permitem escrever programas paralelos. Ex.: Pascal FC, Java, etc.
  - Sistemas Operacionais Paralelos: permitem usar vários processadores em uma máquina. Ex.: Linux, Solaris, Windows, etc.
  - Suportes para Programação Paralela: permitem criar uma máquina paralela virtual. Ex.: PVM

24

### Suporte Computacional

- Sist. Operacionais / Linguagens Multi-tarefa

Máquina

Sistema Operacional / Linguagem Multi-tarefa

Processador

Escalonador

CPU  
N1

25

### Suporte Computacional

- Sistemas Operacionais Paralelos

Máquina paralela

Sistema Operacional Paralelo

Processador 1 ... Processador N

26

### Suporte Computacional

- Suportes para Computação Paralela

Máquina virtual paralela

Linguagem / Suporte para Computação Paralela

Sistema Operacional ... Sistema Operacional

Hardware ... Hardware

27