

Introdução ao Processamento Paralelo e Distribuído



Videoaula

Notas dos slides

APRESENTAÇÃO

O presente conjunto de slides pertence à coleção produzida para a disciplina *Introdução ao Processamento Paralelo e Distribuído* ofertada aos cursos de bacharelado em Ciência da Computação e em Engenharia da Computação pelo Centro de Desenvolvimento Tecnológico da Universidade Federal de Pelotas.

Os slides disponibilizados complementam as videoaulas produzidas e tratam de pontos específicos da disciplina. Embora tenham sido produzidos para ser assistidos de forma independente, a sequência informada reflete o encadeamento dos assuntos no desenvolvimento do conteúdo programático previsto para a disciplina.



2

Modelos de Programação

Para a programação:
Concorrente, Paralela e Distribuída

Notas da videoaula

DESCRIÇÃO

Nesta videoaula é apresentada uma revisão de conceitos associados a processos e threads e sua manipulação no nível do sistema operacional.

OBJETIVOS

Nesta videoaula o aluno reforçará os principais conceitos de sistemas operacionais, tais como processos, threads e escalonamento, necessários ao domínio dos conteúdos trabalhados na disciplina.

4

“ A language that doesn't affect the way you think about programming is not worth knowing.

Alan J. Perlis, desenvolvedor da linguagem ALGOL

Terminologia essencial

Concorrente

Duas ideias: *competição e simultaneidade*. Um programa é descrito em termos de um conjunto de tarefas que disputam acesso aos dados manipulados.

Paralelo

Execução simultânea de tarefas. Quando não há contenção por recursos de hw e não há disputa por dados.

Distribuído

No sistema em execução, decisões sobre a evolução das tarefas são tomadas de forma independente.



5

6

Terminologia essencial

Concorrente

Noção associada à natureza da aplicação. É a aplicação quem determina o grau de independência entre as tarefas, tanto no que diz respeito ao seu ordenamento temporal como na relação de troca de dados.



7

Terminologia essencial

Paralelo

Noção associada às propriedades da arquitetura. Duas execuções são ditas paralelas quando há suporte de hardware suficiente para atender as demandas de ambas.



8

Terminologia essencial

Distribuído

A distribuição prevê um nível de acoplamento, na tomada de decisões, menor. Existe uma pré-disposição a maiores tempos de latência, assim como não é previsto um espaço de endereçamento comum, mas sim uma rede de interconexão para suporte a algum mecanismo de comunicação.



9

Importante saber

Existe uma variedade enorme de ferramentas de programação. Estas ferramentas são voltadas para um ou outro ambiente. É possível combinar o uso de diferentes ferramentas de programação.

Deve se observar que a grande maioria destas ferramentas, além de exigir que o programador identifique a concorrência de sua aplicação, também controle a sincronização e realize o mapeamento da execução do programa sobre os recursos de hardware disponíveis.



10

Importante saber

Outro aspecto relevante é que, na maioria dos casos, a concorrência de uma aplicação é muito maior que o paralelismo suportado pelo hardware.

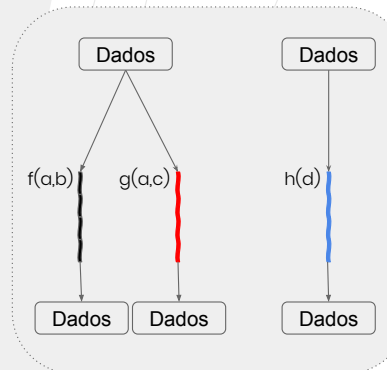
Uma nova geração de ferramentas de programação busca abstrair, pelo menos em parte, a complexidade da gestão dos recursos de hardware.



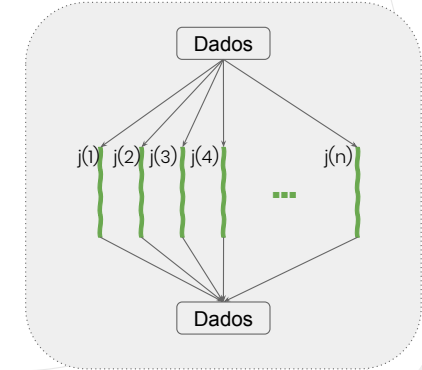
T

Onde Reside a Concorrência

Paralelismo de Tarefas



Paralelismo de Dados



12

Onde Reside a Concorrência

Paralelismo de Tarefas

- As tarefas executam códigos distintos
- Os dados manipulados podem, ou não, ser os mesmos
- As execuções são assíncronas e toda troca de dados entre as tarefas necessita controle externo
- O grau de concorrência é determinado pelo número de tarefas descritas pela aplicação
- O grau de paralelismo é limitado pela quantidade de recursos do hardware

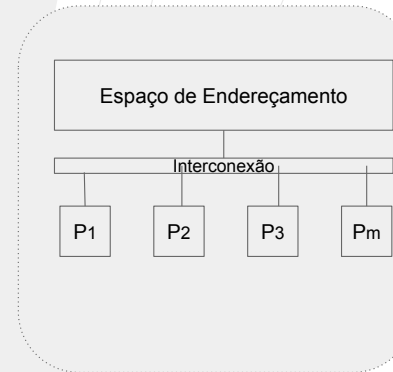
Paralelismo de Dados

- O mesmo código é executado por todas as tarefas
- Cada tarefa manipula um subconjunto dos dados de entrada
- O cálculo é síncrono
- O grau de concorrência é determinado pelo tamanho dos dados de entrada
- O grau de paralelismo é limitado pela quantidade de recursos do hardware

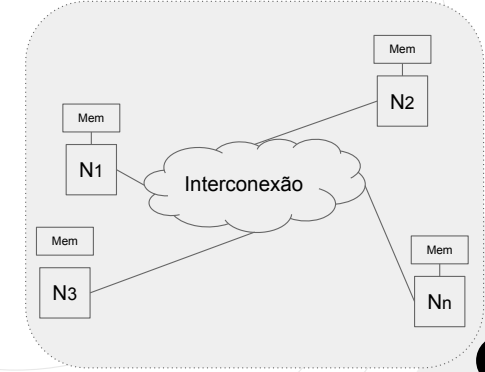
13

Como Explorar o Paralelismo

Multiprocessadores



Multicomputadores



14

Como Explorar o Paralelismo

Multiprocessadores

- Dispõem de espaço de endereçamento compartilhado
- O mecanismo mais eficiente de comunicação envolve operações de leitura e escrita em dados compartilhado
- São necessários mecanismos de sincronização entre os fluxos de execução
- As ferramentas mais adaptadas oferecem recursos para programação multithread

Multicomputadores

- A interconexão de rede deve ser explorada para compartilhamento de dados
- Os dados manipulados podem, ou não, ser os mesmos
- É necessário, de alguma forma, que sejam conhecidos os endereços físicos dos nós participantes em um processamento
- Ferramentas baseadas em RPC/RMI e troca de mensagens, entre outras opções, são as mais populares

15

Ferramentas de Programação

Multiprocessadores

- Pthreads
- OpenMP
- C++ (a partir da versão 11)
- TBB
- Java
- ...

Manycore

- CUDA
- ... (?)

Multicomputadores

- RPC
- RMI (Java)
- MPI (troca de mensagens)
- ...

Heterogêneo

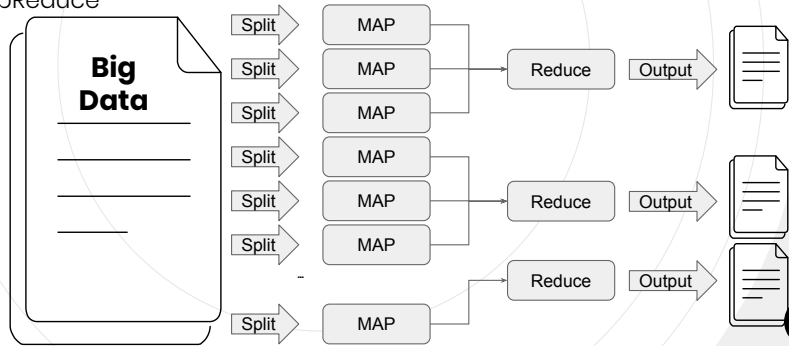
- OpenCL
- SkePU
- ... (?)

16

Ferramentas de Programação

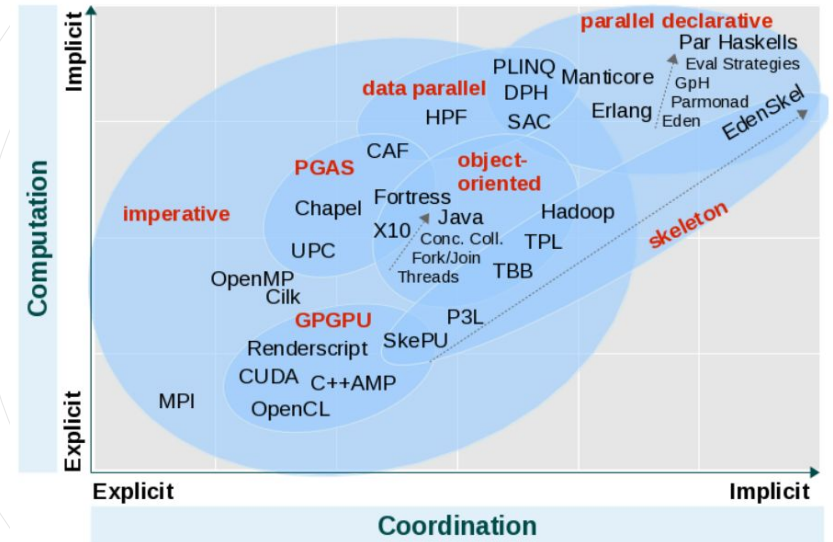
E na nuvem?

- Modelo MapReduce



17

Ferramentas de Programação



18

Fonte: <http://www.macs.hw.ac.uk/cs/techreps/doc0103.html>

Atividade

Em seu ambiente, verifique a disponibilidade de Pthreads e MPI.

Sugestão: utilize ambiente Linux.

Acesse um tutorial sobre Pthreads, como <https://computing.llnwd.net/tutorials/pthread> ou https://www.cin.ufpe.br/~cagf/1f677/2018-1/slides/11_Threads.pdf e execute um programa teste (disponível nestes mesmos links).

Você possui MPI pré-instalado? Se não tiver, verifique possibilidade de instalação em <https://gist.github.com/pajayrao/166bbeaf029012701f790b6943b31bb2>. Procure também compilar e executar um exemplo neste ambiente.



19

Registre suas observações.
<https://forms.gle/zAK58GVrxgpbWELw7>



"A good tool improves the way you work.
A great tool improves the way you think."
Jeff Duntemann

Obrigado!

20