

Programação Concorrente e Paralela

Aleardo Manacero Jr.

Notas Gerais

Objetivos

Os avanços em hardware nos últimos 20 anos trouxe para perto do usuário a possibilidade de programação paralela.

Para aproveitar melhor os recursos de hardware disponíveis, mesmo em celulares, é preciso dominar a programação paralela, tópico desta disciplina.

Com isso, ao final da disciplina espera-se que todos compreendam como tratar problemas de concorrência e consigam produzir aplicações paralelas e de alto desempenho.

Avaliação

- ** Duas Provas (P_1 e P_2) e
- ** Implementação de aplicações concorrentes e paralelas (MProj) e
- ** Produção de artefato de aplicação paralela (opcional)

$$M = \frac{(P_1 + P_2 + MProj)}{3} + Artefato$$

sendo que $0 \leq Artefato \leq 2,0$

Avaliação

Projetos

Projetos serão desenvolvidos em grupos de 2 a 4 estudantes.

Não serão aceitos projetos individuais, nem em grupos de mais de 4 estudantes.

Artefato

O Artefato deve ser uma aplicação paralela, escolhida pelo grupo de alunos e avaliada pelo docente sobre sua adequação.

O grupo deverá produzir o código da aplicação e um artigo acadêmico sobre o trabalho.

Programa

- 1 Princípios básicos de concorrência.
- 2 Soluções para problemas clássicos em memória compartilhada
- 3 Soluções para problemas clássicos em memória distribuída
- 4 Computação de alto desempenho e paralelismo
- 5 Threads e openMP
- 6 Condições e modelos de paralelismo
- 7 Avaliação e otimização de desempenho
- 8 CUDA e openCL
- 9 MPI

Bibliografia

- 1 Andrews, G.; *Foundations of Multithreaded, Parallel and Distributed Programming*, Addison-Wesley, 2000.
- 2 Pacheco, P.S.; *An Introduction to Parallel Programming*, Morgan Kaufmann, 2011.
- 3 Dowd, K. and Severance, C.; *High Performance Computing*, O'Reilly, 1998.
- 4 Ben-Ari, M.; *Principles of Concurrent and Distributed Programming*, Addison-Wesley, 2006.
- 5 Manuais das bibliotecas e linguagens utilizadas na disciplina.

Mais material em <https://www.dcce.ibilce.unesp.br/~aleardo/cursos/pcp>

Previsão de datas

- ① 04/08 Revisão sobre região crítica, semáforos e deadlocks
- ② 08/08 Concorrência usando monitores
- ③ 11/08 Exemplos de problemas concorrentes
- ④ 15/08 Mais exemplos
- ⑤ 18/08 Threads
- ⑥ 22/08 openMP, parte 1
- ⑦ 25/08 openMP, parte 2
- ⑧ 29/08 Concorrência usando troca de mensagens
- ⑨ 01/09 Novos exemplos
- ⑩ 05/09 RPC

Previsão de datas

- ⑪ 15/09 Exemplos
- ⑫ 19/09 Rendezvous
- ⑬ 22/09 Exemplos
- 26/09 Primeira Prova**
- ⑭ 29/09 Computação de alto desempenho
- ⑮ 03/10 GPUs e CUDA
- ⑯ 06/10 CUDA, parte 2
- ⑰ 10/10 Processadores para alto desempenho
- ⑱ 13/10 Memória para alto desempenho
- ⑲ 17/10 Conectividade para alto desempenho

Previsão de datas

- ⑳ 20/10 Condições de paralelismo
- ㉑ 24/10 Modelos de paralelização
- ㉒ 03/11 MPI, parte 1
- ㉓ 07/11 MPI, parte 2
- ㉔ 10/11 Avaliação de desempenho
- ㉕ 14/11 Otimização de código
- ㉖ 17/11 openCL
- 24/11 - Segunda Prova**
- ㉗ 28/11 Apresentações de artefatos

05/12 - EXAME