

MINISTÉRIO DA EDUCAÇÃO

CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA CELSO SUCKOW DA FONSECA

Unidade Descentralizada de Nova Friburgo

CURSO SUPERIOR DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO

DEPARTAMENTO		NOME DA DISCIPLINA				PRÉ-REQUISITOS
		PROGRAMAÇÃO PARALELA E CONCORRENTE				
CÓDIGO		PERÍODO	ANO	SEMESTRE	Programação III	
GSI8605NF		6	2016	II		
CRÉDITOS	AULAS/SEMANA			TOTAL DE AULAS NO SEMESTRE		
	TEÓRICA	PRÁTICA	ESTÁGIO			
3	3	0	0	54		

EMENTA

Princípios e técnicas de programação concorrente
Memória compartilhada e preempção
Exclusão mútua
Semáforos
Monitores
Threads
Projeto de programas paralelos
OpenMP
Comunicação por troca de mensagens
Análise de desempenho
Distribuição de trabalho e recursos
Invocação remota de métodos (RMI)

BIBLIOGRAFIA

Bibliografia básica:

1. SILBERSCHATZ, Abraham. **Fundamentos de Sistemas Operacionais**. 8ª ed. Rio de Janeiro: LTC, 2012.
2. TANENBAUM, Andrew S. **Sistemas Operacionais Modernos**. 3ª ed. São Paulo: Pearson - Prentice Hall, 2010.
3. ANDREWS, G. **Foundations of Multithreaded, Parallel, and Distributed Programming**. Addison-Wesley, 2000.

Bibliografia complementar:

1. TOSCANI, Simão S.; OLIVEIRA, Rômulo da S.; CARISSÍMI, Alexandre da S. **Sistemas Operacionais e Programação Concorrente**. Sagra Luzzatto, 2003. ISBN 9788524106828.
2. HERLIHY, M.; SHAVIT, N. **The Art of Multiprocessor Programming**. Morgan Kaufmann, 2008.
3. SILBERSCHATZ, Abraham; GALVIN, Peter Baer; GAGNE, Greg. **Sistemas Operacionais com Java**. 7ª ed. São Paulo: Campus - Elsevier, 2008.
4. PACHECO, P. **An Introduction to Parallel Programming**. McGraw-Hill, 2011.
5. QUINN, M. **Parallel Programming in C with MPI and OpenMP**. McGraw-Hill, 2003.
6. JAIN, R.. **The Art of Computer Systems Performance Analysis**. Wiley, 1991.

OBJETIVOS GERAIS

Introduzir conceitos e modelos de programação paralela e concorrente.
Possibilitar a experimentação de modelos de programação paralela e concorrente.

METODOLOGIA

Aulas expositivas e dialogadas.
Exercícios com uso de computador.
Trabalhos práticos em grupo.

CRITÉRIO DE AVALIAÇÃO

Avaliação através de provas escritas, com possibilidade de realização de trabalhos práticos.

CHEFE DO DEPARTAMENTO

NOME	ASSINATURA
_____	_____

PROFESSOR RESPONSÁVEL PELA DISCIPLINA

NOME	ASSINATURA
_____	_____

PROGRAMA

1. Princípios e técnicas de programação concorrente
 - Multiprocessadores
 - Memória compartilhada
 - Troca de mensagens
 - Programação paralela e programação concorrente
 - Taxonomia de Flynn
 - Fluxos (*theads*) paralelos
 - Fluxos (*threads*) independentes
 - Processos
 - Preempção
 - Threads cooperativas
 - Exemplos de programas
2. Memória compartilhada e preempção
 - Funcionamento
 - Impasse (*deadlock*)
 - Exclusão mútua
3. Exclusão mútua
 - Eventos
 - Transições
 - Visão como máquinas de estado
 - Estado da *thread*
 - Estado do sistema
 - Travamento (*locks*)
 - Modelos de travamento
 - Região Crítica
 - Algoritmos (Peterson, Filtros, Padaria, etc.)

4. Semáforos
 - Notação de Andrews (atomicidade e *await*)
 - Sincronização
 - Problema do produtor/consumidor
 - Cooperação
 - Buffer limitado
 - Outros exemplos (filósofos, canibais e cozinheiros, etc.)
 - Técnica de passagem de bastão
5. Monitores
 - Conceitos
 - Exemplo – exclusão mútua
 - Cooperação
 - Sinalização
 - Políticas de sinalização
 - Exemplos
 - Cuidados
 - Implementação de monitores
6. Threads
7. Projeto de programas paralelos
 - Oportunidades de paralelização
 - Particionamento
 - Comunicação
 - Aglomeração
 - Mapeamento
8. OpenMP
9. Comunicação por troca de mensagens
 - Troca de mensagens
 - Identificação de pares
 - Semântica de envio e recebimento
 - Formatos e tipos de dados
 - Operações coletivas
 - Exemplos (MPI/OpenMP)
10. Análise de desempenho
 - Modelos de desempenho
 - Tempo de computação
 - Tempo de comunicação
 - Eficiência
 - Aceleração
 - Escalabilidade
 - Projeto de experimentos
 - Métricas
 - Erros comuns de análise de desempenho
 - Medidas de tempos
11. Distribuição de trabalho e recursos
 - Alocação de trabalhos a processadores
 - Distribuição e balanceamento
 - Causas de desbalanceamento
 - Distribuição sob demanda
 - Espaço de tuplas
 - *Pool de threads*
 - Transferência e migração de processos
12. Invocação remota de métodos (RMI)
13. Futuros
14. Memória Transacional