

MINISTÉRIO DA EDUCAÇÃO
CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA CELSO SUCKOW DA FONSECA
DIRETORIA DE ENSINO (DIREN)
DEPARTAMENTO DE ENSINO SUPERIOR (DEPES)
DEPARTAMENTO DE INFORMÁTICA (DEPIN)
CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS PARA INTERNET (CST-SI)

DEPARTAMENTO	PLANO DE CURSO DA DISCIPLINA
DEPIN – Departamento Acadêmico de Informática	ARQUITETURAS AVANÇADAS DE COMPUTADORES

CÓDIGO	PERÍODO	ANO	SEMESTRE	PRÉ-REQUISITOS
GTSI1401	Opt	2014	1	(GTSI1411) Arquitetura de Computadores
CRÉDITOS	AULAS/SEMANA			
2	TEÓRICA	PRÁTICA	ESTÁGIO	TOTAL DE AULAS NO SEMESTRE
	2	0	0	36

EMENTA

Pipelining. Arquiteturas RISC. Processamento paralelo. Computadores multicore. Conceitos Básicos de *Graphics Processing Unit* (GPU).

BIBLIOGRAFIA

Bibliografia básica

1. STALLINGS, W. *Arquitetura e Organização de Computadores*. São Paulo: Pearson.
2. TANENBAUM, A. S. *Organização Estruturada de Computadores*. São Paulo: Prentice-Hall.
3. PATTERSON, D. A., HENNESSY, J. *Arquitetura de Computadores – Uma Abordagem Quantitativa*. 5ª edição. Rio de Janeiro: Campus/Elsevier.

Bibliografia complementar

1. MURDOCCA, M. J., HEURING, V. P. *Introdução à Arquitetura de Computadores*. Rio de Janeiro: Campus/Elsevier.
2. MONTEIRO, M. A. *Introdução à Organização de Computadores*. Rio de Janeiro: LTC.
3. PARHAMI, B. *Arquitetura de Computadores: de Microcomputadores a Supercomputadores*. São Paulo: McGraw-Hill.
4. NULL, L., LOBUR, J. *Princípios Básicos de Arquitetura e Organização de Computadores*. Porto Alegre: Bookman.
5. WEBER, R. F. *Fundamentos de Arquitetura de Computadores*. Porto Alegre: Bookman.

OBJETIVO GERAL

Fornecer o suporte para o entendimento de sistemas computacionais tanto do ponto de vista do software quanto do hardware.

METODOLOGIA

- Aulas expositivas, contando com recursos audiovisuais.
- Aulas eventuais em laboratório de informática, com o uso de um sistema gerenciador de banco de dados relacional de mercado.
- Resolução de exercícios de fixação e propostos.

CRITÉRIO DE AVALIAÇÃO

A avaliação semestral envolve duas provas escritas (P1 e P2). As datas das provas são agendadas entre o professor e a turma. A média parcial (MP) será calculada pelo cômputo da média aritmética simples entre a nota P1 e P2:

$$MP = (P1 + P2) / 2$$

O aluno que faltar a uma das duas provas terá direito a uma avaliação alternativa, denominada segunda chamada, versando sobre todos os tópicos abordados no curso, e cuja data também é agendada entre docente e discentes. A nota obtida nessa 2ª chamada substituirá a da avaliação P1 ou P2 onde o aluno não esteve presente. Caso ele falte às duas avaliações, terá atribuído o grau ZERO em uma delas.

Opcionalmente o docente pode propor um trabalho prático em cada uma das avaliações, com vistas à composição das notas P1 e P2.

Segundo o regimento do CEFET-RJ, caso o aluno obtenha média parcial inferior a 3,0 (três e zero) estará reprovado diretamente. Graus MP maiores ou iguais a 7,0 (sete e zero) aprovam diretamente o aluno. Em situações onde o aluno tenha grau MP entre 3,0 inclusive e 7,0 exclusive, terá direito a uma prova final (PF), que, juntamente com a média parcial gerará uma nova média, denominada média final (MF). Essa média é calculada da seguinte forma:

$$MF = (MP + PF) / 2$$

Para ser aprovado, o aluno deve alcançar uma média final MF maior ou igual a 5,0 (cinco e zero). Caso contrário, estará reprovado, devendo repetir a componente curricular.

PROGRAMA

1. Pipelining
 - 1.1. Conceito e implementação
 - 1.2. Pipeline MIPS
2. Arquiteturas RISC
 - 2.1. Características do conjunto de instruções
 - 2.2. Otimização de registradores
 - 2.3. Pipeline em arquiteturas RISC
3. Processamento paralelo
 - 3.1. Organização de múltiplos processadores
 - 3.2. Multiprocessadores simétricos
 - 3.3. Coerência de cache e protocolo MESI
 - 3.4. *Multithreading* e chips multiprocessadores

- 3.5. Clusters
- 3.6. Acesso não uniforme à memória
- 3.7. Computação vetorial
- 4. Computadores *multicore*
 - 4.1. Desempenho de hardware e software
 - 4.2. Organização *multicore*
- 5. Conceitos Básicos de *Graphics Processing Unit* (GPU)