

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Instituto de Informática - Departamento de Informática Aplicada

Disciplina:	Introdução à Arquitetura de Computadores
Código:	INF01107
Pré-Requisito:	nenhum
Carga Horária:	4 horas aula/semana
Créditos:	04 (quatro)
Semestre:	2009/1
Professores:	Turma A - Raul Fernando Weber Turma B - Carlos Arthur Lang Lisboa Turma C - Ingrid Jansch-Pôrto

Súmula

Representação de dados: sistemas de numeração, aritmética binária e decimal, representação de caracteres. Noções básicas de arquitetura e organização de computadores: organização básica da UCP e variações; conjunto, formato e armazenamento de instruções. Estrutura de software: noções de linguagem assembler; linguagens de programação, compiladores e interpretadores; sistemas operacionais; aplicativos, utilitários e pacotes. Noções rudimentares de entrada e saída, periféricos; comunicação entre máquinas. Prática de laboratório.

Objetivos

Esta disciplina tem por objetivo fornecer o suporte para o entendimento dos conceitos de computação, tanto do ponto de vista de hardware como de software. Assim, através do aprendizado das formas de representação de dados, da estrutura e organização de um computador e de noções básicas de software, o aluno deverá ter, ao final da disciplina, uma visão abrangente da arquitetura e organização de computadores, especialmente nos seguintes pontos:

- Representação de dados: sistemas de numeração, aritmética binária e decimal, representação de números em ponto fixo e ponto flutuante, representação de caracteres.
- Noções básicas de arquitetura e organização de computadores: organização básica da UCP e variações; conjunto de instruções: operações, formato e armazenamento das instruções.
- Conceitos de álgebra booleana. Elementos básicos de hardware. Registradores, unidade aritmética e lógica e memória. Fluxo de dados e de instruções.
- Programação em linguagem de máquina e simbólica. Uso de simuladores de processadores hipotéticos.

Conteúdo Programático

1. Bases numéricas
Conceituação. Métodos de conversão de base: polinomial, por subtrações, por divisões e por substituição.
2. Bases em sistemas de computação
Binária, hexadecimal e octal.
3. Aritmética binária

- Representação de dados (sinal e magnitude, complemento de um, complemento de dois). Soma. Subtração. Conceitos de carry, borrow e overflow.
4. Componentes de um computador: modelo de von Neumann
Unidade central de processamento e memória. Conceito de instrução e formatos. Ciclo de instrução: busca e execução. Conceitos de fluxo de dados e fluxo de controle. Arquiteturas com 4, 3, 2, 1 e 0 endereços.
 5. Processadores hipotéticos: NEANDER e AHMES
Arquitetura: características e conjunto de instruções. Organização e controle. Programação em linguagem de máquina e linguagem simbólica.
 6. Programação nos simuladores NEANDER e AHMES
Uso dos simuladores/depuradores para Neander e Ahmes. Prática de Programação. Exercícios.
 7. Aritmética binária
Multiplicação e divisão.
 8. Números em ponto fixo e ponto flutuante
Representação e operações aritméticas.
 9. Aritmética BCD
Representação, soma e subtração.
 10. Códigos alfanuméricos: representação de informação
Excesso de 3, Código Gray, Códigos de Hamming, ASCII, EBCDIC e Unicode.
 11. Conceitos de álgebra booleana
Operações e propriedades da álgebra booleana. Correlação entre operações da álgebra booleana e os elementos básicos de hardware: portas e inversores. Exemplos de circuitos simples e expressões booleanas correspondentes.
 12. Elementos básicos de hardware
Redes de portas lógicas, multiplexadores e decodificadores. Flip-flops, registradores e contadores. Memória e unidade aritmética e lógica (UAL).
 13. Estudo da organização, elementos de hardware, fluxo de dados e fluxo de execução de instruções em uma máquina com arquitetura de Von Neumann simples.

Total de horas/aula previstas: 60 horas/aula (inclui avaliação)

Técnicas de ensino (experiências de aprendizagem):

A disciplina será desenvolvida através de aulas expositivas e exercícios práticos de programação em linguagem de máquina e linguagem simbólica. Ao longo da disciplina serão exigidos trabalhos de programação, para realização em horário extra-classe, cuja entrega é obrigatória. A disciplina utilizará o sistema de apoio ao ensino Moodle (<http://moodle.inf.ufrgs.br>) para distribuição de material, entrega de trabalhos, organização de grupos de discussão e acompanhamento geral da disciplina.

Todos alunos deverão inscrever-se e atualizar o seu e-mail no sistema adotado e verificar com frequência o site para informações e acompanhamento da disciplina.

Sistema de Avaliação

Provas:

Serão realizadas três provas presenciais, com datas conforme cronograma.

Trabalhos:

Serão realizados 2 trabalhos práticos (TN e TA), usando respectivamente os simuladores Neander e Ahmes. A entrega de todos os trabalhos nos prazos e condições estipuladas é condição necessária para aprovação na disciplina. Os trabalhos devem ser minimamente funcionais, ou seja, devem montar e executar corretamente em um número mínimo de casos de teste. Todos os trabalhos devem ter nota de avaliação superior a zero (0) para que sejam considerados para aprovação.

Divulgação dos resultados:

Os prazos para divulgação dos resultados das provas e trabalhos seguirão os seguintes critérios:

- Provas realizadas no sistema Moodle, sem questões dissertativas: divulgação após a realização das provas por todas as turmas.
- Provas realizadas no sistema Moodle, com questões dissertativas: divulgação em até uma semana após a realização das provas por todas as turmas.
- Provas em papel, sem questões dissertativas: divulgação em até duas semanas após a realização das provas por todas as turmas.
- Provas em papel, com questões dissertativas: divulgação em até três semanas após a realização das provas por todas as turmas.
- Trabalhos com correção automática: divulgação em até duas semanas após a entrega dos trabalhos por todas as turmas
- Trabalhos sem correção automática: divulgação em até quatro semanas após a entrega dos trabalhos por todas as turmas
- Situações imprevistas poderão estender os prazos estabelecidos acima.

Frequência:

De acordo com o regimento da UFRGS, é exigida frequência mínima de 75%.

Conceitos:

O conceito final da disciplina dependerá das notas obtidas nas verificações de aproveitamento e nos trabalhos práticos e da participação em aula, sendo a média final obtida através da fórmula:

$$M = [V1 + V2 + V3 + (TN + 2*TA) / 3] / 4$$

onde V1 e V2 são as notas obtidas nas verificações, e TN e TA, as dos trabalhos práticos Neander e Ahmes, respectivamente. A conversão da média final M para conceitos é feita por meio da seguinte tabela:

9,0 <= M = 10,0: conceito A (aprovado)
7,5 <= M < 9,0: conceito B (aprovado)
6,0 <= M < 7,5: conceito C (aprovado)
4,0 <= M < 6,0: sem conceito (recuperação) podendo passar para conceito C em caso de aprovação ou conceito D em caso de reprovação
0,0 = M < 4,0: conceito D (reprovado)
Não entrega dos 2 trabalhos práticos (funcionais): conceito D (reprovado)
Faltas > 25%: conceito FF (reprovado)

Atividades de recuperação:

Recuperação por motivo de saúde: de acordo com o regimento da Universidade, através de processo aberto na Junta Médica da UFRGS, o aluno poderá recuperar as provas ou os trabalhos em data, horário e local a serem marcados pelo professor.

Recuperação de média insuficiente: o aluno com média inferior a 6 mas superior a 4, e que tiver entregue todos os trabalhos da disciplina poderá recuperar o conceito realizando uma prova versando sobre todo o conteúdo do programa, que substitui a menor nota entre as 3 provas. Não há recuperação dos trabalhos.

Bibliografia Básica

- Weber, R. F. **Fundamentos de Arquitetura de Computadores, 3ª edição**. Porto Alegre: Instituto de Informática da UFRGS, 2004. Ed. Artmed, Série de Livros Didáticos do Instituto de Informática da UFRGS, volume 8, 2008.

Software Utilizado na Disciplina

Simuladores de arquiteturas hipotéticas (Neander, Ahmes) especialmente desenvolvidos para os trabalhos da disciplina.