



## Plano de Ensino

### 1. Dados de Identificação

Turma: 13280 - Ciência da Computação - 4ª Fase - Noturno - 2016/1

Comp. Curricular: GEX100 - Organização de computadores

Número de Créditos: 4

Carga horária - Hora Aula: 72

Carga horária - Hora Relógio: 60

Professor: Adriano Sanick Padilha

Horário de Atendimento: Quarta-feira – 19:00 às 21:00h.

### 2. Objetivo Geral do Curso

O curso tem por objetivo a formação integral de novos cientistas e profissionais da computação, os quais deverão possuir conhecimentos técnicos e científicos e serem capazes de aplicar estes conhecimentos, de forma inovadora e transformadora, nas diferentes áreas de conhecimento da Computação. Adicionalmente, os egressos do curso deverão ser capazes de adaptar-se às constantes mudanças tecnológicas e sociais, e ter uma formação ao mesmo tempo cidadã, interdisciplinar e profissional

### 3. Ementa

Tendências tecnológicas na fabricação de CPUs e memórias. CPU: instruções e modos de endereçamento. Formatos de instruções e linguagem de montagem. Simulador e montador. Datapath e unidade de controle. Alternativas de implementação (monociclo, multiciclo, pipeline, superescalar). Exceções e interrupções. Conflitos estruturais, de dados e de controle. Hierarquia de memória e associatividade (cache e TLB). Dispositivos de entrada e saída: tipos, características e sua conexão à CPU e à memória. Comunicação com a CPU (polling, interrupção, DMA).

### 4. Objetivo

#### 4.1 Geral

Conhecer os principais conceitos envolvidos na concepção da arquitetura e organização dos computadores, sabendo caracterizar e entender o funcionamento interno de um computador, dada uma certa arquitetura, reconhecendo o impacto de cada alternativa sobre fatores como desempenho e custo.

#### 4.2 Específicos

- Apresentar os conceitos fundamentais de um computador em termos de seus componentes básicos (processador, sistema de memória e dispositivos de entrada e saída), abstraindo sua implementação física;
- Prover exemplos reais e contemporâneos desses componentes básicos
- Estabelecer a noção de modelo de programação (programmer's view) de um sistema computacional;
- Mostrar o papel da linguagem de montagem como formato intermediário para geração de código .
- Prover uma visão panorâmica da cadeia de ferramentas de programação de sistemas (compilador, montador, ligador, carregador, simulador do conjunto de instruções e depurador).

*Adriano*



5. Cronograma e Conteúdo Programático

	Conteúdo	Horas Acumuladas
1	Apresentação do Plano de Ensino; Visão geral da disciplina.	4
2	ORGANIZAÇÃO DE UM COMPUTADOR - Componentes básicos de um computador; - O papel da tecnologia de circuitos integrados no projeto de um computador; - Tendências tecnológicas na construção de computadores.	8
3	EVOLUÇÃO E DESEMPENHO DO COMPUTADOR - Arquitetura x86; - Sistemas Embarcados e a ARM; - Medida e métrica de desempenho; - Programas para avaliação de desempenho (“benchmarks”); - Formas de comparação de desempenho.	12
4	O SUBSISTEMA DE MEMÓRIA - Classificação de memórias; - A organização hierárquica de memória e seu gerenciamento; - Memórias cache: <ul style="list-style-type: none"><li>• Uso da localidade espacial, projeto de um sistema de memória para suportar cache;</li><li>• Políticas de mapeamento, atualização e consistência;</li><li>• Organização em múltiplos níveis;</li><li>• Caches associativas e tamanho de rótulos, seleção do bloco a ser substituído;</li><li>• Impacto no desempenho.</li></ul> - Organização da MEM cache do ARM.	18
5	MEMÓRIA VIRTUAL: - Paginação, segmentação, fragmentação; - Colocação de uma página na memória principal, faltas de página, escrita de uma página; - TLB, integração de memória virtual, TLBs e caches, operação completa da hierarquia de memória, proteção com memória virtual; - Memória virtual: manipulação de faltas de página e de faltas na TLB. Uma estrutura comum para hierarquias de memória;	24
6	O SUBSISTEMA DE ENTRADA E SAÍDA (E/S) - Tipos e características de dispositivos de E/S; - Conexão de dispositivos de E/S com processador e memória; - Interfaceamento de dispositivos de E/S com a memória, o processador e o sistema operacional:	30

Adriano

	<ul style="list-style-type: none"> <li>• "Polling";</li> <li>• Via interrupções;</li> <li>• Acesso direto à memória (DMA).</li> </ul> <p>- Interface Processador/Periféricos: introdução, impacto do sistema de E/S no desempenho, medidas de desempenho de E/S;</p> <p>- Exemplo de dispositivos de E/S contemporâneos;</p>	
7	<p><b>SUPOORTE AO SISTEMA OPERACIONAL</b></p> <ul style="list-style-type: none"> <li>- Visão Geral do SO;</li> <li>- Escalonamento;</li> <li>- Gerenciamento de Memória;</li> <li>- Gerenciamento de Memória no ARM.</li> </ul>	34
8	<p><b>AVALIAÇÃO DE NP1</b></p>	36
9	<p><b>ARITMÉTICA DO COMPUTADORES</b></p> <ul style="list-style-type: none"> <li>- Unidade Lógica e Aritmética;</li> <li>- Representação de inteiros e sua aritmética;</li> <li>- Representação de ponto flutuando e sua aritmética.</li> </ul>	40
10	<p><b>CONJUNTO DE INSTRUÇÕES</b></p> <ul style="list-style-type: none"> <li>- Características e Funções: <ul style="list-style-type: none"> <li>• Características da instruções de máquina;</li> <li>• Tipo de operandos;</li> <li>• Tipo de operações;</li> <li>• Tipo de operações do ARM.</li> </ul> </li> <li>- Modos e Formatos de Endereçamento: <ul style="list-style-type: none"> <li>• Endereçamento;</li> <li>• Endereçamento do ARM;</li> <li>• Formato de instruções;</li> <li>• Formato de instruções do ARM;</li> <li>• Linguagem de montagem.</li> </ul> </li> </ul>	48
11	<p><b>O PROCESSADOR: UNIDADES DE PROCESSAMENTO E CONTROLE</b></p> <ul style="list-style-type: none"> <li>- Estrutura e comportamento de unidades de processamento (Ups);</li> <li>- Organização de uma UP com instruções mono-ciclo;</li> <li>- Organização de uma UP com instruções multi-ciclo;</li> <li>- Comportamento da unidade de controle (UC) para uma dada UP;</li> <li>- Suporte ao tratamento de exceções na UP e na UC;</li> <li>- Pipeline de Instruções;</li> <li>- Processador ARM;</li> </ul>	56

*Adriano*



12	<p>PROGRAMAÇÃO DE SISTEMAS</p> <p>- Representações de código:</p> <ul style="list-style-type: none"><li>• Linguagem de alto nível;</li><li>• Linguagem de montagem;</li><li>• Linguagem de máquina;</li><li>• Anatomia de arquivos-objeto e arquivos executáveis.</li></ul> <p>- A cadeia de ferramentas para geração e inspeção de código:</p> <ul style="list-style-type: none"><li>• A estrutura de um compilador;</li><li>• O mecanismo interno de um montado;</li><li>• Ligador: mecanismos estático e dinâmico;</li><li>• Funções de um carregador e de um simulador;</li><li>• O mecanismo interno de um depurador.</li></ul> <p>- Aplicações e casos de uso do ARM:</p>	64
13	<p>SEMINÁRIO:</p> <p>- MODELO DE PROGRAMAÇÃO DO SISTEMA (“programmer’s view”);</p> <p>- SUBPROGRAMAÇÃO;</p> <p>- EXCEÇÕES E INTERRUPÇÕES.</p>	68
14	<p>APRESENTAÇÃO DE TRABALHOS DE NP2.</p>	72

## 6. Procedimentos Metodológicos (estratégias de ensino, equipamentos, entre outros)

Aulas expositivas com recursos multimídia e quadro/gis, discussões sobre artigos de revistas técnicas (tecnológicas) em sala de aula e utilização de simuladores para a contextualização do conteúdo teórico exposto. O professor utilizará o ambiente virtual MOODLE como ambiente de ensino-aprendizagem, nele o aluno encontrará artigos, slides das aulas expositivas e lista de exercícios.

Horário de atendimento aos acadêmicos será nas quartas-feira no período vespertino (19:00h-21:00h).

## 7. Avaliação do Processo Ensino-Aprendizagem

A avaliação será realizada através de provas escritas, trabalhos e seminários. A composição da na nota final (NF) é a média aritmética das notas parciais (NP):  $NF = (NP1 + NP2 + NP) / 3$ . O acadêmico terá a aprovação da disciplina se a sua NF for igual ou superior a 6.

A composição das NPs são formadas pelas médias ponderadas dos trabalhos práticos (60%) e o seminário de apresentação (40%) do assunto pertinente. Caso o acadêmico não atinja uma NP igual ou superior a 6, será realizada uma avaliação de recuperação (AR) contemplando todo o conteúdo trabalhado.

### 7.1 Processo de recuperação da nota de avaliação

Na aula subsequente ao avaliação parcial, a mesma será discutida em aula, esclarecendo as dúvidas pertinentes as questões. Nesta aula será definida a data da avaliação de recuperação. A avaliação de recuperação substituirá a nota da avaliação parcial na integralização da NF, como explicado no item 7.

*Adriano*



## 8. Referências

### 8.1 Básicas

**PATTERSON**, David A.; **HENNESSY**, John L.. Organização e Projeto de Computadores. Rio de Janeiro, RJ, Elsevier, 2005.

**STALLINGS**, William. Arquitetura e Organização de Computadores: projeto para o desempenho. 5a ed., São Paulo, SP, Prentice Hall, 2005.

**TANENBAUM**, Andrew S.. Organização Estruturada de Computadores. 5a ed., Rio de Janeiro, Prentice-Hall do Brasil, 2006.

### 8.2 Específicas

**MURDOCCA**, Miles J.. Introdução à Arquitetura de Computadores. Rio de Janeiro, Campus, 2001.

**HENNESSY**, John L.. Arquitetura de Computadores: uma abordagem quantitativa. Rio de Janeiro, Campus, 2003.

**MANO**, M.. Computer System Architecture. Englewood Cliffs, NJ, Prentice-Hall International, 1993.

**HEURING**, Vincent P.. Computer Systems Design and Architecture. 2a ed., Upper Saddle River, NJ, Pearson Prentice Hall, 2004.

**HARRIS**, David Money. Digital Design and Computer Architecture. Amsterdam, Elsevier, 2007.

*Marco A. Spohn*  
**MARCO AURÉLIO SPOHN**  
Siape nº. 1521671  
Coord. do Curso de Ciência da Computação  
Universidade Federal da Fronteira Sul-UFFS  
Campus Chapecó-SC

*Adriano S. P. Dill*  
PROFESSOR