

## Plano de Ensino

---

### 1) Identificação

**Disciplina:** INE5411 - Organização de Computadores I  
**Turma(s):** 03208A, 03208B, 03208C  
**Carga horária:** 108 horas-aula Teóricas: 92 Práticas: 16  
**Período:** 1º semestre de 2011

### 2) Cursos

- Ciências da Computação (208)

### 3) Requisitos

- INE5406 - Sistemas Digitais

### 4) 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. Aritmética. Avaliação de desempenho. Datapath e unidade de controle. Alternativas de implementação (monociclo, multiciclo, pipeline, superescalar). Exceções e interrupções. Hazards 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).

### 5) Objetivos

**Geral:** Definir a interface binária entre hardware e software e sua relação com os utilitários binários (montador e ligador) e o núcleo do sistema operacional, além de quantificar o impacto da organização do computador em seu desempenho.

**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.
- Prover uma visão panorâmica da cadeia de ferramentas de programação de sistemas (compilador, montador, ligador, carregador, simulador do conjunto de instruções e depurador).
- Mostrar o papel da linguagem de montagem como formato intermediário para geração de código.
- Codificar pequenos programas na linguagem de montagem de um processador escolhido e executá-los em um simulador de seu conjunto de instruções.

### 6) Conteúdo Programático

#### 6.1) ORGANIZAÇÃO DE UM COMPUTADOR [5 horas-aula]

- Componentes básicos de um computador.
- O papel da tecnologia de circuitos integrados no projeto de um computador.
- Tendências tecnológicas na construção de computadores.

#### 6.2) DESEMPENHO EM UM COMPUTADOR [7 horas-aula]

- Medida e métrica de desempenho.
- Programas para avaliação de desempenho ("benchmarks").
- Formas de comparação de desempenho.
- Exemplo de desempenho de processadores contemporâneos.

#### 6.3) A ARQUITETURA DO CONJUNTO DE INSTRUÇÕES DE UM PROCESSADOR [13 horas-aula]

- Suporte a operandos em linguagens de programação
  - Operandos escalares (variáveis, constantes) e estruturas de dados.
  - Tipos de dados e suas consequências ("endian" e alinhamento).
- Suporte a operações e construções em linguagens de programação
  - Instruções aritméticas, lógicas, de comparação e de deslocamento.

- Instruções para tomadas de decisão em construções condicionais e laços.
  - Instruções para suporte a subrotinas (procedimentos, funções e métodos).
  - Instruções para suporte ao paradigma de orientação a objetos.
- Modos de endereçamento
    - Direto em registrador.
    - Base.
    - Imediato.
    - Relativo ao PC.
    - Modos compostos ou derivados.
  - Representação de instruções em linguagem de máquina.
  - Pseudo-instruções.
  - Exemplos de instruções de arquiteturas contemporâneas.
  - Exemplos de código em diferentes níveis de representação.
- 6.4) EFEITOS DE REPRESENTAÇÃO NUMÉRICA EM ARITMÉTICA INTEIRA [3 horas-aula]
- Representação de números sinalizados e não sinalizados.
  - Overflow e extensão de sinal.
- 6.5) O PROCESSADOR: UNIDADES DE PROCESSAMENTO E CONTROLE [13 horas-aula]
- Estrutura e comportamento de unidades de processamento (UPs).
  - Organização de uma UP (mono-ciclo).
  - Comportamento da unidade de controle (UC) para uma dada UP.
  - Suporte ao tratamento de exceções na UP e na UC.
  - Exemplos de organização de processadores contemporâneos.
- 6.6) ACELERAÇÃO COM A TÉCNICA “PIPELINE” [11 horas-aula]
- Hazards estruturais, de dados e de controle.
  - Impacto dos hazards no desempenho.
  - Organização de uma UP com pipeline e respectiva UC.
  - Paralelismo em nível de instruções (ILP).
  - Mecanismos básicos de exploração de ILP.
    - Despacho múltiplo: superescalar e "very long instruction word" (VLIW)
    - Execução especulativa.
    - Escalonamento estático e dinâmico.
  - Exemplo de organização de pipeline em processador contemporâneo.
- 6.7) O SUBSISTEMA DE MEMÓRIA [11 horas-aula]
- Classificação de memórias.
  - A organização hierárquica de memória e seu gerenciamento.
  - Memórias cache.
    - Políticas de mapeamento, atualização e consistência.
    - Associatividade.
    - Organização de um controlador de cache (multi-ciclo).
    - Organização em múltiplos níveis.
    - Impacto no desempenho.
  - Memória virtual e suporte de hardware para tradução de endereços
    - "Translation Lookaside Buffer"(TLB).
  - Exemplos de subsistemas de memória contemporâneos.
- 6.8) O SUBSISTEMA DE ENTRADA E SAÍDA (E/S) [9 horas-aula]
- 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
    - "Polling".
    - Via interrupções.
    - Acesso direto à memória (DMA).
  - Exemplo de dispositivos de E/S contemporâneos.
- 6.9) PROGRAMAÇÃO DE SISTEMAS [9 horas-aula]
- Representações de código:
    - Linguagem de alto nível
    - Linguagem de montagem
    - Linguagem de máquina
    - Anatomia de arquivos-objeto e arquivos executáveis.
  - A cadeia de ferramentas para geração e inspeção de código.
    - A estrutura de um compilador.
    - O mecanismo interno de um montador.
    - Ligador: mecanismos estático e dinâmico.

- Funções de um carregador e de um simulador.
  - O mecanismo interno de um depurador.
  - Aplicações e casos de uso de linguagens de montagem.
- 6.10) MODELO DE PROGRAMAÇÃO DO SISTEMA (“programmer’s view”) [9 horas-aula]
- Registradores e memória.
  - Conjunto de instruções e modos de endereçamento.
  - Uso de memória (segmento de dados, segmento de pilha, segmento de código).
  - E/S mapeada em memória.
- 6.11) SUBPROGRAMAÇÃO [9 horas-aula]
- Convenção de chamada.
  - Salvamento e recuperação de contexto.
  - Layout da pilha e chamadas recursivas.
- 6.12) EXCEÇÕES E INTERRUPÇÕES [9 horas-aula]
- Registradores de controle.
  - Tratamento de exceções e interrupções

## 7) Metodologia

### 7.1. Instrumentos metodológicos

A metodologia de ensino consiste na exposição dos principais conceitos de organização de computadores e programação de sistemas de acordo com o livro-texto adotado na bibliografia básica. Os principais conceitos são ilustrados, em sala de aula, através de exemplos e estudos de caso. Para fixação dos conceitos, o professor propõe um conjunto de exercícios a serem resolvidos pelos alunos.

A familiarização com ferramentas de programação de sistemas é obtida a partir de aulas de laboratório.

Como instrumento adicional, haverá um horário semanal de atendimento extra-classe (com duração total de 2 horas), mediante agendamento prévio.

Aos alunos que queiram aprofundar-se em tópicos avançados, sugere-se a leitura dos textos da bibliografia complementar.

Nas aulas de laboratório, prevê-se o emprego de estagiário de docência, de acordo com as normas da UFSC e do INE, dependendo da disponibilidade de aluno de pós-graduação interessado.

### 7.2 Pressupostos da metodologia

A metodologia adotada pressupõe que os alunos de um curso diurno não se limitam a comparecer às aulas, mas utilizam um número de horas, no mínimo igual ao número de horas-aula, para as atividades extra-classe associadas a esta disciplina (leitura, resolução de exercícios e prática de programação em linguagem de montagem).

Pressupõe-se que os alunos tenham estudado todas as páginas indicadas do livro-texto e tenham resolvido, como atividade extra-classe, todos os exercícios propostos pelo professor.

Para garantir a máxima concentração dos alunos durante as aulas e garantir a lisura do processo de avaliação, em nenhuma aula (expositiva, laboratório, ou avaliação) é permitido o uso de dispositivos pessoais móveis (celulares, *smartphones*, *tablets*, *MP3 players* e similares).

Como cópias dos slides são disponibilizadas na *webpage* da disciplina, não será necessário anotar o conteúdo das aulas. Para assegurar máxima concentração durante as aulas expositivas, não é permitido o uso de computadores pessoais (*desktops*, *notebooks*, *palmtops* ou similares).

Nas aulas de laboratório (apesar de a sala alocada possuir computadores do tipo *desktop*), é permitido o uso de *notebook* próprio, desde que seu uso seja restrito à atividade prevista para cada aula.

Durante as avaliações é permitida a consulta apenas ao material fornecido pelo professor. O professor avisará os alunos em quais avaliações é permitido o uso de calculadora.

## 8) Avaliação

### 8.1. Instrumentos de avaliação

A avaliação da aprendizagem é realizada através de quatro provas escritas e um conjunto de relatórios de aulas de laboratório.

As três primeiras (**P1**, **P2** e **P3**) são provas regulares e a última (**REC**) é uma prova de recuperação (conforme art. 70, § 2º, da Resolução 17/CUn/97).

A cada laboratório corresponde um relatório a ser entregue ao final da aula. Os relatórios podem ser realizados por grupos de no máximo dois alunos. A ausência do(a) aluno(a) implica em lhe ser atribuída nota zero no respectivo relatório. O(A) aluno(a) que chegar atrasado(a) às aulas de laboratório (fora do intervalo de tolerância estabelecido pelo professor) ou se ausentar dela antes de concluir os experimentos terá sua nota no

respectivo relatório multiplicada por 0,5 para refletir sua participação parcial.

A avaliação global das aulas de laboratório será representada por uma nota **LB**, calculada como a média aritmética simples das notas obtidas nos relatórios.

Os tópicos do conteúdo programático serão assim avaliados:

- P1: Tópicos 1-5 e 9-10
- P2: Tópicos 1-6 e 9-11
- P3: Tópicos 1-8 e 9-12
- LB: Tópicos 9-12
- REC: Tópicos 1-8 e 9-12

Para a realização das quatro provas, serão alocadas 08 horas-aula da carga da disciplina.

A nota final (NF) será calculada como função da frequência do aluno e das notas por ele obtidas nas provas e nos relatórios, conforme formalizado a seguir.

### 8.2. Critérios para aprovação ou reprovação

- a) O aluno que não comparecer a no mínimo 75% das aulas será considerado reprovado por frequência insuficiente (FI), de acordo com o artigo 73, do Capítulo I, Seção IX do Regimento Geral da UFSC. Neste caso, **NF = 0,0**.
- b) O critério para aprovação ou reprovação dos alunos com frequência suficiente (FS) baseia-se na média final (MF) assim calculada: **MF = 0,2.P1 + 0,3.P2 + 0,4.P3 + 0,1.LB**.
- c) Será considerado aprovado o aluno com FS e  $MF \geq 6,0$ . Neste caso, **NF = MF**.
- d) Será considerado reprovado o aluno com FS e  $MF < 3$ . Neste caso, **NF = MF**.

### 8.3. Mecanismo de substituição de prova perdida

Se o(a) aluno(a) faltar a uma das provas regulares por motivo justificável, devidamente comprovado, deverá requerer junto à Secretaria do INE, no prazo de 72 horas, a autorização para substituir a prova, através de uma prova substitutiva única (**PS**), cujo conteúdo avaliado será equivalente ao conteúdo global da(s) prova(s) perdida(s).

Decorrido o prazo sem qualquer requerimento, será atribuída nota zero à prova perdida. Se a justificativa for julgada procedente pelo INE, o(a) aluno(a) fica automaticamente convocado(a) a fazer a prova substitutiva (**PS**). Neste caso, a(s) nota(s) da(s) prova(s) perdida(s) será(ão) substituída(s) pela nota obtida em **PS** e será então efetuado o cálculo de MF, conforme o item 8.2b. No caso de FI, o aluno perderá o direito de fazer a prova **PS**, recaindo-se no caso do item 8.2.a.

Como uma aula de laboratório é insubstituível, também é insubstituível a nota atribuída ao relatório de uma aula de laboratório perdida, à qual será atribuída nota zero.

Conforme parágrafo 2º do artigo 70 da Resolução 17/CUn/97, o aluno com frequência suficiente (FS) e média final no período (**MF**) entre 3,0 e 5,5 terá direito a uma nova avaliação ao final do semestre (**REC**), sendo a nota final (**NF**) calculada conforme parágrafo 3º do artigo 71 desta resolução, ou seja: **NF = (MF + REC) / 2**.

## 9) Cronograma

O cronograma de aulas e avaliações será publicado (e atualizado conforme a necessidade) na seguinte URL: <http://www.inf.ufsc.br/~santos/ine5411/aulas.htm>.

Entretanto, as datas das provas não sofrerão mudança, salvo determinação em contrário de superior hierárquico ou alterações no calendário escolar. Ficam designadas as seguintes datas para as provas:

- P1: 03 de maio de 2011.
- P2: 07 de junho de 2011.
- P3: 05 de julho de 2011.
- PS: 12 de julho de 2011.
- REC: 14 de julho de 2011.

As provas serão realizadas sempre no horário das 10:10 às 11:50.

## 10) Bibliografia Básica

- David A. Patterson and John L. Hennessy, "Computer Organization and Design: The Hardware/Software Interface", 4rd edition, Morgan Kaufmann Publishers, San Francisco, California, USA, 2010.

## 11) Bibliografia Complementar

- John L. Hennessy and David A. Patterson, "Computer Architecture: A Quantitative Approach", 4th edition, Morgan Kaufmann Publishers, San Francisco, California, 2003.
- Dominic Sweetman, "See MIPS Run", Morgan Kaufmann Publishers, San Francisco, California, 1999.