

Plano de Ensino

1) Identificação

Disciplina: INE5411 - Organização de Computadores I
Turma(s): 03208A, 03208C, 06235
Carga horária: 108 horas-aula Teóricas: 92 Práticas: 16
Período: 1º semestre de 2024

2) Cursos

- Ciências da Computação (208)
- Engenharia Eletrônica (235)

3) Requisitos

- Ciências da Computação (208)
 - INE5406 - Sistemas Digitais
- Engenharia Eletrônica (235)
 - EEL7030 - Microprocessadores
 - INE5406 - Sistemas Digitais

4) Professores

- Luiz Claudio Villar dos Santos (luiz.santos@ufsc.br)

5) 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).

6) 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.

7) Conteúdo Programático

- 7.1) 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.
- 7.2) 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.
- 7.3) 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.
- 7.4) EFEITOS DE REPRESENTAÇÃO NUMÉRICA EM ARITMÉTICA INTEIRA [3 horas-aula]
- Representação de números sinalizados e não sinalizados.
 - Overflow e extensão de sinal.
- 7.5) 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.
- 7.6) 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.
- 7.7) 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.
- 7.8) 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.
- 7.9) 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.
- 7.10) **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.
- 7.11) **SUBPROGRAMAÇÃO** [9 horas-aula]
- Convenção de chamada.
 - Salvamento e recuperação de contexto.
 - Layout da pilha e chamadas recursivas.
- 7.12) **EXCEÇÕES E INTERRUPÇÕES** [9 horas-aula]
- Registradores de controle.
 - Tratamento de exceções e interrupções

8) Metodologia

8.1. 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 práticas baseadas em um **simulador** que executa pequenos programas tal como ocorreria em um processador real.

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 do texto indicado na bibliografia complementar.

8.2 Pressupostos da metodologia

A metodologia adotada pressupõe que os discentes não se limitem às aulas presenciais, mas utilizem parte da carga da disciplina para autoestudo (leitura e resolução de exercícios). Se necessário, recomenda-se o uso de horas adicionais para garantir plena cobertura do texto e dos exercícios.

Pressupõe-se que os alunos tenham estudado as páginas indicadas do livro-texto e tenham resolvido, como atividades complementares, os exercícios propostos pelo professor.

Durante as avaliações é permitida a consulta apenas ao material fornecido pelo professor. Nas avaliações não será permitido o uso de calculadora nem de qualquer outro dispositivo eletrônico.

9) Avaliação

9.1. Instrumentos de avaliação

A avaliação da aprendizagem é realizada através de três provas escritas (**P1**, **P2** e **P3**) e um conjunto de relatórios de aulas de laboratório.

Estão previstas entre 6 e 8 aulas de laboratório. A cada laboratório corresponde um relatório a ser entregue até o final da aula. Será atribuída nota zero a um relatório não entregue.

As notas serão graduadas de 0 (zero) a 10 (dez), não podendo ser fracionadas aquém ou além de 0,5, conforme o art. 71 da Resolução 17/Cun/97.

A avaliação global das aulas de laboratório será representada por uma nota **LB**, calculada como a média aritmética das notas obtidas nos relatórios, previamente arredondadas conforme o art. 71, § 1o, antes de aplicar a média.

Os tópicos do conteúdo programático serão assim passíveis de avaliação:

- 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

Para a realização das três provas, serão alocadas 06 horas-aula da carga da disciplina.

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

9.2. Critérios para aprovação ou reprovação

- a) Os estudantes que não obtiverem presença correspondente ao mínimo de 75% das aulas serão considerados reprovados 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 de estudantes 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) Serão considerados aprovados estudantes com FS e MF $\geq 6,0$. Neste caso, **NF = MF**.
- d) Serão considerados reprovados estudantes com FS e MF < 3 . Neste caso, **NF = MF**.

9.3. Mecanismo de substituição de prova perdida

Se faltar a uma das provas regulares por motivo justificável, devidamente comprovado, a(o) estudante deverá requerer junto à Secretaria do INE, no prazo de 3 dias úteis, a autorização para realizar uma prova substitutiva. Haverá 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) estudante 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 9.2.b. No caso de FI, a(o) estudante perderá o direito de fazer a prova **PS**, recaindo-se no caso do item 9.2.a.

9.4 Mecanismo de Recuperação

A recuperação (**REC**) será realizada através de uma prova escrita, onde todo o conteúdo programático será passível de avaliação (tópicos de 1 a 12). Para a realização da prova de recuperação, serão alocadas 2 horas-aula da carga da disciplina.

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**.

10) Cronograma

O cronograma de aulas e avaliações será publicado (e atualizado conforme a necessidade) no sistema Moodle. Seguem as semanas prováveis para realização das provas:

- P1: Semana 7.
- P2: Semana 11.
- P3: Semana 15.
- PS: Semana 15.
- REC: Semana 17.

11) Bibliografia Básica

- David A. Patterson e John L. Hennessy, “Organização e Projeto de Computadores : a Interface Hardware/Software”, tradução da 5a. edição, Elsevier, 2017 (ISBN 978853528793).

12) Bibliografia Complementar

- David A. Patterson and John L. Hennessy, “Computer Organization and Design: The Hardware/Software Interface”, 5th edition, Elsevier (Morgan Kaufmann), USA, 2014 (ISBN 978-0-12-407726-3).
- Dominic Sweetman, “See MIPS Run”, Second Edition, Morgan Kaufmann, Elsevier, 2005 (ISBN 978-0-12-088421-6).