



MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA

PROGRAMA DE ENSINO

1. Identificação

Disciplina: INE5411 - Organização de Computadores I
Nível: Graduação
Carga Horária: 108 horas-aula (Teórica: 92; Prática: 16)
Vigência: De 2023-1 até a presente data

2. 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).

3. Cursos Relacionados

- CIÊNCIAS DA COMPUTAÇÃO (208) - Currículo: 2007-1 (Obrigatória)
 - ENGENHARIA ELETRÔNICA (235) - Currículo: 2009-2 (Obrigatória)
-

4. Objetivos

4.1 Objetivo 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.

4.2 Objetivos Específicos:

- a) 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.
- b) Prover exemplos reais e contemporâneos desses componentes básicos.
- c) Estabelecer a noção de modelo de programação (programmer's view) de um sistema computacional.
- d) Prover uma visão panorâmica da cadeia de ferramentas de programação de sistemas (compilador, montador, ligador, carregador, simulador do conjunto de instruções e depurador).

- e) Mostrar o papel da linguagem de montagem como formato intermediário para geração de código.
 - f) Codificar pequenos programas na linguagem de montagem de um processador escolhido e executá-los em um simulador de seu conjunto de instruções.
-

5. Conteúdo Programático

- 1 ORGANIZAÇÃO DE UM COMPUTADOR [5 horas-aula]
 - 1.1 Componentes básicos de um computador.
 - 1.2 O papel da tecnologia de circuitos integrados no projeto de um computador.
 - 1.3 Tendências tecnológicas na construção de computadores.
- 2 DESEMPENHO EM UM COMPUTADOR [7 horas-aula]
 - 2.1 Medida e métrica de desempenho.
 - 2.2 Programas para avaliação de desempenho (?benchmarks?).
 - 2.3 Formas de comparação de desempenho.
 - 2.4 Exemplo de desempenho de processadores contemporâneos.
- 3 A ARQUITETURA DO CONJUNTO DE INSTRUÇÕES DE UM PROCESSADOR [13 horas-aula]
 - 3.1 Suporte a operandos em linguagens de programação
 - 3.1.1 Operandos escalares (variáveis, constantes) e estruturas de dados.
 - 3.1.2 Tipos de dados e suas conseqüências ("endian" e alinhamento).
 - 3.2 Suporte a operações e construções em linguagens de programação
 - 3.2.1 Instruções aritméticas, lógicas, de comparação e de deslocamento.
 - 3.2.2 Instruções para tomadas de decisão em construções condicionais e laços.
 - 3.2.3 Instruções para suporte a subrotinas (procedimentos, funções e métodos).
 - 3.2.4 Instruções para suporte ao paradigma de orientação a objetos.
 - 3.3 Modos de endereçamento
 - 3.3.1 Direto em registrador.
 - 3.3.2 Base.
 - 3.3.3 Imediato.
 - 3.3.4 Relativo ao PC.
 - 3.3.5 Modos compostos ou derivados.
 - 3.4 Representação de instruções em linguagem de máquina.
 - 3.5 Pseudo-instruções.
 - 3.6 Exemplos de instruções de arquiteturas contemporâneas.
 - 3.7 Exemplos de código em diferentes níveis de representação.
- 4 EFEITOS DE REPRESENTAÇÃO NUMÉRICA EM ARITMÉTICA INTEIRA [3 horas-aula]
 - 4.1 Representação de números sinalizados e não sinalizados.
 - 4.2 Overflow e extensão de sinal.
- 5 O PROCESSADOR: UNIDADES DE PROCESSAMENTO E CONTROLE [13 horas-aula]
 - 5.1 Estrutura e comportamento de unidades de processamento (UPs).
 - 5.2 Organização de uma UP (mono-ciclo).
 - 5.3 Comportamento da unidade de controle (UC) para uma dada UP.
 - 5.4 Suporte ao tratamento de exceções na UP e na UC.
 - 5.5 Exemplos de organização de processadores contemporâneos.
- 6 ACELERAÇÃO COM A TÉCNICA ?PIPELINE? [11 horas-aula]

- 6.1 Hazards estruturais, de dados e de controle.
- 6.2 Impacto dos hazards no desempenho.
- 6.3 Organização de uma UP com pipeline e respectiva UC.
- 6.4 Paralelismo em nível de instruções (ILP).
- 6.5 Mecanismos básicos de exploração de ILP.
 - 6.5.1 Despacho múltiplo: superescalar e "very long instruction word" (VLIW)
 - 6.5.2 Execução especulativa.
 - 6.5.3 Escalonamento estático e dinâmico.
- 6.6 Exemplo de organização de pipeline em processador contemporâneo.
- 7 O SUBSISTEMA DE MEMÓRIA [11 horas-aula]
 - 7.1 Classificação de memórias.
 - 7.2 A organização hierárquica de memória e seu gerenciamento.
 - 7.3 Memórias cache.
 - 7.3.1 Políticas de mapeamento, atualização e consistência.
 - 7.3.2 Associatividade.
 - 7.3.3 Organização de um controlador de cache (multi-ciclo).
 - 7.3.4 Organização em múltiplos níveis.
 - 7.3.5 Impacto no desempenho.
 - 7.4 Memória virtual e suporte de hardware para tradução de endereços
 - 7.4.1 "Translation Lookaside Buffer"(TLB).
 - 7.5 Exemplos de subsistemas de memória contemporâneos.
- 8 O SUBSISTEMA DE ENTRADA E SAÍDA (E/S) [9 horas-aula]
 - 8.1 Tipos e características de dispositivos de E/S.
 - 8.2 Conexão de dispositivos de E/S com processador e memória.
 - 8.3 Interfaceamento de dispositivos de E/S com a memória, o processador e o sistema operacional
 - 8.3.1 "Polling".
 - 8.3.2 Via interrupções.
 - 8.3.3 Acesso direto à memória (DMA).
 - 8.4 Exemplo de dispositivos de E/S contemporâneos.
- 9 PROGRAMAÇÃO DE SISTEMAS [9 horas-aula]
 - 9.1 Representações de código:
 - 9.1.1 Linguagem de alto nível
 - 9.1.2 Linguagem de montagem
 - 9.1.3 Linguagem de máquina
 - 9.1.4 Anatomia de arquivos-objeto e arquivos executáveis.
 - 9.2 A cadeia de ferramentas para geração e inspeção de código.
 - 9.2.1 A estrutura de um compilador.
 - 9.2.2 O mecanismo interno de um montador.
 - 9.2.3 Ligador: mecanismos estático e dinâmico.
 - 9.2.4 Funções de um carregador e de um simulador.
 - 9.2.5 O mecanismo interno de um depurador.
 - 9.3 Aplicações e casos de uso de linguagens de montagem.
- 10 MODELO DE PROGRAMAÇÃO DO SISTEMA (?programmer?s view?) [9 horas-aula]
 - 10.1 Registradores e memória.
 - 10.2 Conjunto de instruções e modos de endereçamento.
 - 10.3 Uso de memória (segmento de dados, segmento de pilha, segmento de código).

- 10.4 E/S mapeada em memória.
 - 11 SUBPROGRAMAÇÃO [9 horas-aula]
 - 11.1 Convenção de chamada.
 - 11.2 Salvamento e recuperação de contexto.
 - 11.3 Layout da pilha e chamadas recursivas.
 - 12 EXCEÇÕES E INTERRUPÇÕES [9 horas-aula]
 - 12.1 Registradores de controle.
 - 12.2 Tratamento de exceções e interrupções
-

6. Bibliografia Básica

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

7. Bibliografia Complementar

- [1] 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).
- [2] Dominic Sweetman, ?See MIPS Run?, Second Edition, Morgan Kaufmann, Elsevier, 2005 (ISBN 978-0-12-088421-6).