

MC404 - Organização de Computadores e Linguagem de Montagem  
1º semestre de 2012 - Turma E - [Prof. Ricardo Anido](#)

[Aulas](#) | [Avisos](#) | [Bibliografia](#) | [Plano](#) | [Laboratórios](#) | [Ferramentas e outros recursos](#) | [Notas](#)

Horários de aula

Dia	Horário	Salas
Quinta-Feira	19:00 - 21:00	CB05
Sexta-feira	21:00 - 23:00	LM03

Horários de atendimento

Fernando Thiers (fernandothiers arroba gmail ponto com) (PAD): agendar por e-mail (sala a combinar)

Prof. Ricardo Anido (ranido arroba ic ponto unicamp ponto br) agendar por e-mail (Sala 37 do IC 1)

Critérios de Avaliação

Os alunos serão avaliados através de duas provas escritas e de aproximadamente 10 trabalhos de laboratório.

Datas das provas escritas:

- P1 (26/4)
- P2 (21/6)
- Exame (12/7)

Seja:

- P1 e P2 as notas das Provas 1 e 2, respectivamente.
- MP a média das Provas.
- MTr a média dos M trabalhos de laboratório.
- MMC404 a média da disciplina MC404
- NFinal a nota Final de MC404

Onde:

- $MP = (4P1 + 6P2) / 10$
- MTr = Média ponderada dos M trabalhos de laboratório; os primeiros quatro trabalhos terão peso 2, os demais peso 3.
- Cálculo da média MMC404:
  - Se  $MP > 5,0$  e  $MTr > 5,0$ ,  $MMC404 = (6*MP + 4*MTr)/10$
  - Caso contrário,  $MMC404 = \text{MIN}(MP, MTr)$

A nota final (NFinal) será:

- $NFinal = MMC404$ , se  $MMC404 \geq 5,0$
- $NFinal = (MMC404 + Exame) / 2$ , caso contrário

OBS: Qualquer tentativa de fraude implica em NFinal=0 e reprovação.

(\*)Toda submissão de atividade ou trabalho de laboratório deverá respeitar as regras definidas na atividade. As regras definirão os arquivos a serem enviados bem como o prazo para submissão.

A qualidade e apresentação do programa serão levadas em conta na nota da atividade.

## Programa da Disciplina

- Introdução à arquitetura de computadores digitais. Componentes básicos de uma CPU: UC, ULA, endereçamento, registradores; dispositivos de E/S; fases da execução de uma instrução.
- Revisão de números binários e operações aritméticas; notação 1-complemento e 2-complemento; notação hexadecimal; caracteres ASCII; números em ponto flutuante.
- Introdução a um processador didático (Faíska): instruções básicas, linguagem de montagem.
- Repertório de instruções do Faíska: instruções aritméticas, lógicas, desvios, deslocamento. Exemplos de programação em linguagem de montagem com essas instruções.
- Procedimentos, passagem de parâmetros, variáveis locais a procedimentos.
- Entrada e saída, acesso a dispositivos de E/S. Interrupções.
- Memórias - tipos e hierarquias
- Introdução à arquitetura de processador real: ARM. repertório de instruções e modos de endereçamento.
- Montador e macro-montador para o processador ARM. linguagem de montagem do ARM.
- E/S no ARM: registradores de E/S; interrupções; temporizadores; watch-dog timer. técnicas para gravação de código no ARM a partir de um PC.
- Exemplos de programação em linguagem de montagem do ARM: aritmética, E/S, interrupções, temporizadores.
- Integração entre programação em C e em linguagem de montagem: técnicas, cuidados; compilador gcc para o ARM
- Introdução ao ambiente de programação ARM linguagem C; depurador; execução passo a passo.
- Biblioteca do gcc para o ARM; exemplos de uso. Exercícios de programação em C com o ARM.
- Introdução a ligadores: ligação estática e ligação dinâmica.
- Noções de arquiteturas de outros processadores de 32 bits: MIPS, PowerPC.

## Bibliografia

- Apostila (processador didático Faíska)
  - [Capítulo 1 - Representação de informação na memória](#) (pdf)
  - [Capítulo 2 - Organização de computadores e introdução a linguagens de montagem](#) (pdf)
  - [Capítulo 3 - O processador Faíska](#) (pdf)
  - [Capítulo 4 - Procedimentos e Funções](#) (pdf)
  - [Capítulo 5 - Entrada e saída](#) (pdf)
  - [Folha de Referência Processador Faíska \(resumo dos comandos\)](#) (pdf)
- Livros
  - *Organização Estruturada de Computadores*  
A. S. Tanenbaum, Prentice-Hall, 5a edição, 2006.
  - *Computer Organization and Design Fourth Edition : The Hardware/Software Interface*  
David A. Patterson, John L. Hennessy, Morgan Kaufmann, 2009.
  - *A Programmer's View of Computer Architecture*  
J. Goodman e K. Miller, Oxford University Press