

Aula 1

Informações Gerais

MAC0216 - Técnicas de Programação I

Professores: Alfredo, Daniel, Fabio e Kelly

Departamento de Ciência da Computação
Instituto de Matemática e Estatística



1.

O que veremos
em MACo216?

Objetivos

Expor conceitos e ambientes de programação e integração de módulos e programas

- ▷ Início: baixo nível (linguagem de montagem)
- ▷ Meio: uso de ferramentas de
 - sistema operacional
 - desenvolvimento de software
- ▷ Fim: princípios de orientação a objetos

Tópicos abordados I

- ▷ Arquitetura de computadores
- ▷ Linguagem de montagem e ligação de código objeto
- ▷ Interação com o sistema operacional e Shell scripts
- ▷ Gerenciamento de compilação de programas
- ▷ Modularização de código
- ▷ Bibliotecas estáticas e dinâmicas

Tópicos abordados II

- ▷ Teste e depuração
- ▷ Controle de versão
- ▷ Linguagens procedimentais de alto nível (C)
- ▷ Linguagens dinâmicas de script (Python)
- ▷ Introdução aos conceitos de orientação a objetos (em Python)

The Missing Semester of Your CS Education (disciplina do MIT)

<https://missing.csail.mit.edu/>

Classes teach you all about advanced topics within CS, from operating systems to machine learning, but there's one critical subject that's rarely covered, and is instead left to students to figure out on their own: proficiency with their tools. We'll teach you how to master the command-line, use a powerful text editor, use fancy features of version control systems, and much more!

Students spend hundreds of hours using these tools over the course of their education (and thousands over their career), so it makes sense to make the experience as fluid and frictionless as possible. Mastering these tools not only enables you to spend less time on figuring out how to bend your tools to your will, but it also lets you solve problems that would previously seem impossibly complex.

2.

Qual será o formato
do curso?

Formato do curso

- ▷ Aulas online síncronas
- ▷ Videoaulas previamente gravadas
- ▷ Palestras
- ▷ Leituras recomendadas
- ▷ Exercícios semanais (individual)
 - Quizzes
 - Exercícios-programas
- ▷ 4 Projetos (em grupo)
 - Exercícios-programas mais complexos

Materiais didáticos

- ▷ Slides das aulas
- ▷ Vídeos
- ▷ Leituras indicadas
 - Livros
 - Páginas web
 - Artigos técnicos ou científicos
- ▷ Códigos de exemplos

Ferramentas

- ▷ e-Disciplinas
 - Disponibilização de materiais didáticos
 - Entrega de tarefas
 - Comunicação (avisos, fórum de dúvidas)
- ▷ Google Meet
 - Aulas síncronas
- ▷ IDEs, compiladores, interpretadores, controladores de versão, ...
 - Para os exemplos e exercícios-programas

3.

Como a avaliação
será feita?

Critérios de avaliação

- ▷ Nota de participação no curso (NC)
- ▷ Média das notas dos exercícios semanais (ME)
- ▷ Média das notas dos projetos (MP)
- ▷ Cálculo da Média Final (MF)

$$MF = (2NC + 3ME + 5MP) / 10$$

- ▷ Se $MF \geq 5$, você estará aprovado
- ▷ Se $3 \leq MF < 5$, você estará de recuperação
- ▷ Se $MF < 3$, você estará reprovado
- ▷ Presença = 100% **mas esteja presente nas aulas!**

4.

Como ser bem sucedido
em MACo216?

Dicas de ouro

- ▶ Participe de forma ativa no curso
 - Interaja com os professores e monitor
 - Use o fórum de dúvidas do e-Disciplinas
 - Forme grupos de estudo com os colegas
- ▶ Reserve tempo para fazer as atividades do curso tão logo elas sejam propostas
 - Não acumule as tarefas!
- ▶ Entregue códigos-fonte que sejam agradáveis de se ler

Código de ética da USP

<http://www.leginf.usp.br/?resolucao=resolucao-no-4871-de-22-de-outubro-de-2001-3>

Artigo 23 – É vedado aos membros do corpo docente e demais alunos da Universidade:

I – prolongar indevidamente o período de formação acadêmica ou manter matrícula com o objetivo de utilizar as estruturas da Universidade;

II – lançar mão de meios e artifícios que possam fraudar a avaliação do desempenho, seu ou de outrem, em atividades acadêmicas, culturais, artísticas, desportivas e sociais, no âmbito da Universidade, e acobertar a eventual utilização desses meios.

Referências Bibliográficas

Básica:

- ▷ W.R. Stevens, S.A. Rago, Advanced Programming in the UNIX Environment (3rd Edition), Addison-Wesley Professional, 2013.
- ▷ B. Neveln, LINUX assembly language programming, Prentice Hall, 2000.
- ▷ B.W. Kernigham and R. Pike, The practice of programming, Addison-Wesley, 1999.
- ▷ N. Nisan, S. Schocken, The Elements of Computing Systems: Building a Modern Computer from First Principles, MIT Press, 2005.
- ▷ E. Roberts, The Art and Science of C, Addison-Wesley, 1995.
- ▷ A. Martelli, Python in a Nutshell, 2nd edition. O'Reilly, 2006.

Complementar:

- ▷ Intel 64 and IA-32 Architectures Software Developers' Manual, https://en.wikibooks.org/wiki/X86_Assembly/GAS_Syntax.
- ▷ GNU Make Manual, <http://www.gnu.org/software/make>.
- ▷ Git Documentation, <https://git-scm.com/doc>.

MACo216

Técnicas de Programação I

Professores

- ▶ Alfredo Goldman - gold@ime.usp.br
- ▶ Daniel Batista - batista@ime.usp.br
- ▶ Fabio Kon - kon@ime.usp.br
- ▶ Kelly R. Braghetto - kellyrb@ime.usp.br

Monitor

- ▶ Bruno Cunha