

MC102 – Algoritmos e Programação de Computadores

Plano de Desenvolvimento da Disciplina Primeiro Semestre de 2020

Este plano foi preparado de acordo com o Programa Emergencial criado pela [Resolução 25/2020](#).

Turmas	Professor(a)	Página com informações específicas
ABC	Pedro Felipe do Prado	https://sites.google.com/view/mc102abc2020
EF	Jacques Wainer	http://www.ic.unicamp.br/~wainer/cursos/1s2020/102/topo102.html
GHI	Sandra Avila	http://www.ic.unicamp.br/~sandra/teaching/2020-1-mc102ghi
KLMN	Zanoni Dias	https://www.ic.unicamp.br/~zanoni/mc102/2020-1s
OXZ	Julio Cesar Lopez Hernandez	https://sites.google.com/site/unicampjlopez
4567	Arthur Valencio	http://www.arthurvalencio.com/mc102
Coordenação		Página com informações gerais
	Islene Calciolari Garcia	http://www.ic.unicamp.br/~mc102

Ementa

Conceitos básicos de organização de computadores. Construção de algoritmos e sua representação em pseudocódigo e linguagens de alto nível. Desenvolvimento sistemático e implementação de programas. Estruturação, depuração, testes e documentação de programas. Resolução de problemas.

Linguagem de Programação

Python, versão 3.

Critério de Avaliação

Serão levados em consideração os seguintes elementos:

- **Tarefas de Laboratório:** programas a serem implementados e submetidos à correção automática via [SuSy](#);
- **Provas mediadas por tecnologia:** provas com questões a serem respondidas via Moodle na área [MC102 - 1S2020](#).

Tarefas de Laboratório

Serão propostas n tarefas de laboratório, que deverão ser implementadas pelos(as) alunos(as). Os programas desenvolvidos serão testados com um jogo de testes pré-determinado, subdividido em *testes abertos*, que podem ser vistos pelos(as) alunos(as) e *testes fechados*, que não podem ser vistos. A nota de cada laboratório será proporcional ao número de testes, abertos ou fechados, que executaram corretamente. No entanto, serão considerados válidos apenas os programas projetados de maneira a resolver o problema proposto para um conjunto amplo de possibilidades e não aqueles projetados para emitir a saída correta para o subconjunto de testes abertos.

Juntamente com o enunciado de cada tarefa de laboratório L_i será indicado o peso desta tarefa $PL_i \in \{1, 2, 3, 4\}$. A média das tarefas de laboratório, M_L , é a média ponderada destas n notas.

O gerenciamento da submissão e testes das tarefas de laboratório é feito com o auxílio de um sistema automatizado conhecido como **SuSy** (Submission and Testing System for Student Programs), e está acessível a partir do endereço web <https://susy.ic.unicamp.br:9999/mc102>.

Provas mediadas por tecnologia

Serão aplicadas duas provas: P_1 com peso 3 e P_2 com peso 4. Estas avaliações serão realizadas via Moodle na área [MC102 - 1S2020](#). As provas estarão abertas por 48 horas para **todas as turmas** nas datas abaixo:

P_1	P_2
07 a 08 de maio	25 a 26 junho

A média das avaliações presenciais, M_p , é a média ponderada das provas P_1 e P_2 considerando seus respectivos pesos.

Cálculo da Média Final

A média final M_{Final} e a situação do(a) aluno(a) serão definidas de acordo com as regras a seguir. De acordo com o Regimento Geral de Graduação os(as) alunos(as) devem ter frequência **Freq** maior ou igual a 75% para aprovação, mas durante o período de suspensão das atividades presenciais não haverá controle de presenças e faltas.

- Caso $M_p \geq 5$ e $M_L \geq 5$:

O(A) aluno(a) estará **aprovado(a) por nota e frequência** com Média Final:

$$M_{Final} = 0.7 * M_p + 0.3 * M_L$$

- Caso o(a) aluno(a) não esteja aprovado(a) pela regra anterior e $M_p \geq 2.5$ e $M_L \geq 2.5$:

Deverá ser realizado um exame mediado por tecnologia que estará aberto por 72 horas para **todas as turmas** nas datas abaixo:

Exame
13 a 15 de julho

O cálculo da Média Final será feito tendo como base a Média Preliminar, como descrito abaixo:

$$M_{Preliminar} = \min(4.9, 0.7 * M_p + 0.3 * M_L)$$

$$M_{Final} = (M_{Preliminar} + \text{Exame})/2$$

- Caso $M_{Final} \geq 5.0$ o(a) aluno(a) estará **aprovado(a) por nota e frequência**.
 - Caso contrário, estará **reprovado(a) por nota**.
- Caso $M_p < 2.5$ ou $M_L < 2.5$:

O(A) aluno(a) estará **reprovado(a) por nota** com $M_{Final} = \min(M_p, M_L)$.

Atendimento

Este oferecimento de MC102 conta com a colaboração de vários alunos de pós-graduação (PEDs) e graduação (PADs) que atuam como monitores(as). Durante o período de suspensão das atividades presenciais o atendimento será feito via Fóruns no Moodle na área [MC102 - 1S2020](#). Atendimentos extras serão divulgados na página web da disciplina e/ou nas páginas web específicas das turmas. No eventual retorno às atividades presenciais, o atendimento será descrito em <http://www.ic.unicamp.br/~mc102/horarios.html>.

Moodle

A área [MC102 - 1S2020](#) do sistema Moodle está operante e conterà o espaço para realização das provas, os fóruns para atendimentos e exercícios extras. Mais informações serão divulgadas na página web da disciplina.

Informações Adicionais:

- Todas as atividades avaliativas são individuais. Qualquer tentativa de fraude implicará em nota **0.0 (zero)** na disciplina para **todas** as pessoas envolvidas.
- Fraudes nas tarefas de laboratório poderão ser detectadas automaticamente entre **todas** as respostas (de **todos(as)** os(as) alunos, entre **todas** as turmas) ao longo do semestre.
- A submissão de um código que não produz as saídas corretas dos testes via a implementação dos algoritmos solicitados nas tarefas de laboratório, mas que exibe as saídas esperadas dos testes abertos a partir da comparação de trechos da entrada será considerada fraude.

Material de Apoio

Serão indicados vídeos para cada semana do curso de acordo com o [Cronograma de exploração de conteúdos](#) disponível no site da disciplina.

- Recomendamos as páginas oficiais da linguagem Python: www.python.org ou www.python.org.br (em português).
- Indicamos também o livro *How to Think Like a Computer Scientist: Interactive Edition* de Brad Miller e David Ranum, disponível em inglês e português, conforme descrito a seguir:
 - How to Think Like a Computer Scientist: Interactive Edition. <<https://runestone.academy/runestone/static/thinkcspy/index.html>>
 - Aprendendo com Python: Edição interativa (usando Python 3.x) <<https://panda.ime.usp.br/pensepy/static/pensepy/index.html>>

Outros textos e alguns vídeos serão indicados na página web geral da disciplina. Caso consulte algum material, verifique se a versão de Python descrita é a 3.x. Se não for, fique atento(a) aos detalhes que variam de uma versão para outra.
