



ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO
Departamento de Engenharia de Computação e Sistemas Digitais

PCS 2190 – FUNDAMENTOS DE COMPUTAÇÃO I

2º. Semestre / 2018

(versão 1.0: 10/07/2018)

Professor: [Ricardo Nakamura](#) ricardonakamura@usp.br

Horários de Aula: 3ª. feira, 20:50 às 22:30 -

Atendimento: 3ªs. Feiras, das 20:00 às 20:50

Website (TIDIA-Ae): <http://ae4.tidia-ac.usp.br/>

Mensagens anônimas: <http://pcs2190.sarahah.com>

Objetivos

Introduzir os estudantes no campo de conhecimento da computação aplicada ao design, abordando seus aspectos teóricos e práticos.

Introduzir a discussão sobre as características das interfaces homem-computador.

Introduzir a lógica de programação.

Método de Avaliação

A média final **MF** é calculada pela média aritmética de três notas: duas provas teóricas e uma nota atribuída de acordo com o desempenho dos alunos nas atividades desenvolvidas em sala de aula e em laboratório.

$$\mathbf{MF} = (\mathbf{P}_1 + \mathbf{P}_2 + 2*\mathbf{A}) / 3$$

P1 = Prova 1 (0 a 10)

P2 = Prova 2 (0 a 10)

A = Acompanhamento das atividades (0 a 5)

Bibliografia Principal

SHIFFMAN, D. **Learning Processing: A Beginner's Guide to Programming Images, Animation, and Interaction**; Morgan Kaufmann, 2008, 450 p.

REAS, C.; FRY, B. **Processing: a programming handbook for visual designers and artists**. MIT Press. 2007. 737 p.

PUGA, S.; RISSETI, G. **Lógica de Programação e Estruturas de Dados**. 2ª ed. Prentice-Hall Brasil, 2008. 264p.



ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO
Departamento de Engenharia de Computação e Sistemas Digitais

Bibliografia Complementar:

MAEDA, J. **Maeda & Media**. Nova York: Rozzoli, 2000.

GREENBERG, I. **Processing: Creative Coding and Computational Art**. Friends of Ed, 2007. 840 p.

NEGROPONTE, N. **A vida digital**. São Paulo: Companhia das Letras, 1995.

WHITE, R.; DOWNS, T.E. **How Computers Work**. Que, 2003.

WONG, W.; WONG, B. **Diseño gráfico digital**. Barcelona: Gustavo Gili, 2004.

Apostilas:

TORI, R.; NAKAMURA, R. Fundamentos de Computação I.

CRONOGRAMA

AGOSTO

21 (1) Apresentação da disciplina / Algoritmos: Conceitos (sala de aula)

28 (2) Programação: algoritmos imperativos (sala de aula)

SETEMBRO

04 **RECESSO**

11 (3) Programação: repetição e condicional com Português Estruturado (sala de aula)

18 (4) Programação usando Scratch (1) (sala de aula)

25 (5) Programação usando Scratch (2) (laboratório)

OUTUBRO

02 (6) Programação usando Scratch (3) (laboratório)

09 (7) **PROVA 1** (sala de aula)

16 (8) Programação usando Processing (laboratório)

23 (9) Programação: Animação (laboratório)

30 (10) Programação: Exercícios (laboratório)

NOVEMBRO

06 (11) Programação: Vetores (laboratório)

13 (12) Programação: Exercícios (laboratório)

20 **RECESSO**

27 (13) Programação: Exercícios (laboratório)

DEZEMBRO

04 (14) **PROVA 2** (sala de aula)

11 (15) **PROVA SUB** (sala de aula)

18 Semana de TCC 2 do curso de Design



ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO
Departamento de Engenharia de Computação e Sistemas Digitais

PROGRAMA AULA A AULA

Avaliações que integram a nota A indicadas em itálico.

Aula	Tópico	Atividades
1	Apresentação da disciplina Algoritmos: Conceitos	Apresentação expositiva sobre objetivos e métodos da disciplina e critérios de avaliação. Discussão sobre expectativas dos alunos e conhecimento prévio sobre programação. Apresentação expositiva sobre algoritmos. Atividades práticas sobre a execução de algoritmos.
2	Programação: algoritmos imperativos	Atividades práticas com ênfase na solução de problemas lógicos e na execução de algoritmos imperativos.
3	Programação: repetição e condicional com Portugues Estruturado	Antes da aula: assistir vídeo sobre algoritmos e português estruturado. A: <i>Teste sobre conceitos expostos no vídeo.</i> Atividades práticas em papel, com ênfase na interpretação e execução de algoritmos.
4	Programação usando Scratch (1)	Antes da aula: assistir vídeo tutorial de desenvolvimento de programa em Scratch. A: <i>Teste sobre conceitos expostos no vídeo.</i> Atividades de programação utilizando o Scratch.
5	Programação usando Scratch (2)	Antes da aula: fazer a lista de exercícios práticos. A: <i>Teste sobre algoritmos.</i> Atividades práticas de programação utilizando o Scratch, com ênfase no uso de condicionais.
6	Programação usando Scratch (3)	Antes da aula: fazer a lista de exercícios práticos. A: <i>Teste sobre algoritmos.</i> Atividades práticas de programação utilizando o Scratch, com ênfase no uso de repetição.
7	PROVA 1	Prova individual dissertativa, em sala de aula, com questões sobre algoritmos, utilizando as notações adotadas até o momento.
8	Programação usando Processing	Antes da aula: assistir vídeo de apresentação da ferramenta Processing. A: <i>Teste sobre conceitos expostos no vídeo.</i>



ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO
Departamento de Engenharia de Computação e Sistemas Digitais

Aula	Tópico	Atividades
		Atividades práticas para transição da programação usando Scratch para a programação usando Processing.
9	Programação: Animação	Antes da aula: estudo programado (Animação) A: <i>Teste sobre algoritmos.</i> Atividades práticas de programação utilizando recursos do Processing para criar animações.
10	Revisão geral	Realização de lista de exercícios práticos e atendimento a dúvidas sobre o que foi estudado até o momento. A: <i>Acompanhamento da participação dos alunos.</i>
11	Programação: Vetores	Antes da aula: estudo programado (Vetores) A: <i>Teste sobre conceitos expostos no vídeo.</i> Atividades práticas de programação envolvendo manipulação de vetores para desenhos.
12	Programação: Exercícios	Antes da aula: fazer a lista de exercícios práticos. A: <i>Teste sobre algoritmos.</i> Atividades práticas de programação envolvendo manipulação de vetores para animações.
13	Programação: Exercícios	Antes da aula: fazer a lista de exercícios práticos. A: <i>Teste sobre algoritmos.</i> Atividades práticas de programação combinando os diversos conceitos vistos durante o semestre.
14	PROVA 2	Prova individual dissertativa, em sala de aula, com questões sobre programação usando a notação do Processing.
15	PROVA SUB	Prova substitutiva para alunos que não fizeram a Prova 1 ou Prova 2. Formato: individual dissertativa, com questões sobre algoritmos e programação.