

# Algoritmos e Lógica de Programação

IAL002

## Professor

-  Lucas Baggio Figueira
-  Seg & Sex 17h às 18h
-  Teams
-  <http://fatecrp.edu.br>
-  lucas.figueira@fatec.sp.gov.br

## Info do Curso

-  Pré-requisito: Nenhum
-  Terças-feiras
-  19h às 22h30m
-  Laboratório 1 ou Sala 7

## Sobre

A compreensão e construção de algoritmos são fundamentais para a Ciência da Computação bem como para toda a Tecnologia da Informação, sobretudo na Sociedade da Informação hiper-conectada onde problemas e oportunidades surgem no ambiente virtual como reflexo da intersecção com o mundo real os algoritmos se tornam protagonistas como ferramenta para proposição de soluções computacionais para as mais diversas áreas do conhecimento.

## Ementa

Projeto e representação de algoritmos. Estruturas de controle de fluxo de execução: seqüência, seleção e repetição. Tipos de dados básicos e estruturados (vetores e registros). Rotinas. Arquivos. Implementação de algoritmos usando uma linguagem de programação.

## Objetivos

Analisar problemas computacionais e projetar soluções por meio da construção de algoritmos.

## Métricas de Avaliação

Teremos 3 (três) avaliações (*AV*) no semestre, denominadas  $Av_1$ ,  $Av_2$  e  $Av_3$ , valendo [0, 10], sendo que cada avaliação terá sua particularidade e está descrita abaixo:

- $Av_1$  - Prova manuscrita contendo questões para produção de algoritmos e/ou programas.
- $Av_2$  - Prova manuscrita contendo questões de execução de algoritmos e/ou programas por meio do teste de mesa.
- $Av_3$  - Prova digitada contendo questões de construção de programas no computador.

Teremos também atividades em laboratório e extra-laboratório, valendo [0, 10], que serão avaliadas e comporão a nota final.

- $At_L$  - Atividades em laboratório.
- $At_E$  - Atividades extra-laboratório.

O cálculo da nota final (*NF*) será composto por 90% pelas avaliações (*AV*) e 10% pelas atividades (*AT*), portanto o cálculo será dado pela fórmula:

$$\begin{aligned} AV &= \frac{Av_1 + Av_2 + Av_3}{3} \\ AT &= \frac{At_L + At_E}{2} \\ NF &= \frac{9AV + AT}{10} \end{aligned}$$

Serão considerados aprovados aqueles que obtiverem  $NF \geq 6$ , sendo que aqueles cuja a  $NF < 6$  poderão realizar uma avaliação substitutiva com o objetivo de sobrepor a menor nota obtida nas avaliações, e assim aumentar sua *NF*.

Aviso aos estudantes em Regime Especial: Considerar  $AT = At_E$ .

## Referências Bibliográficas

### Básico

1. ASCENCIO, A. F. G. CAMPOS, E. A. V. *Fundamentos da Programação de Computadores: algoritmos, Pascal e C/C++ e Java*. Longman, 2007.
2. FORBELLONE, L. V., EBERSPACHER, H. F. *Lógica de Programação: A Construção de algoritmos e estruturas de dados*. Prentice Hall, 2005.
3. ZIVIANI, Nívio. *Projeto de Algoritmos com Implementações em Pascal e C*. 2 ed. Thomson Pioneira, 2004.

### Complementar

1. ASSIS, C. *Introdução à Programação: 500 Algoritmos Resolvidos*. LTC, 2002.
2. MANZANO, J. A., OLIVEIRA, J. F. *Algoritmos: Lógica Para Desenvolvimento de Programação de Computadores*. 2a. ed. Érica, 2019.
3. BHARGAVA, A. Y. *Entendendo Algoritmos: Um Guia Ilustrado Para Programadores e Outros Curiosos*. Novatec, 2017.

# FAQs

?

Haverá programação de computadores nessa disciplina?

!

Sim, com certeza, usaremos a linguagem C++ para implementar os algoritmos construídos e assim verificarmos seu funcionamento por parte do computador.

?

É necessário saber micro-informática, ou seja saber "mexer" no computador?

!

Sim, embora não seja um pré-requisito é importante ter boas noções de micro-informática, mas não é impedimento desde que o estudante saiba "se virar". Lembrando que nesta disciplina não abordaremos absolutamente nada sobre micro-informática, operação de computadores e seus sistemas operacionais (Windows, Linux, macOS, etc.)

?

É preciso ter tido contato com programação previamente?

!

Não, de forma alguma.

?

Esta disciplina é fundamental para o curso?

!

Sim, esta disciplina está diretamente relacionada com todas as disciplinas de programação do curso, e como está logo no primeiro ciclo ela é a base de todas as subsequentes na área de programação

## Código de Conduta Ética

Em avaliações não aceite ou peça assistência de qualquer natureza.

As avaliações individuais, como o próprio nome diz, não permitem a interferência de agentes externos na resolução do problema. São considerados agentes externos, ou assistência de qualquer natureza, orientações, discussões, apoio em qualquer estratégia e/ou codificação que não seja planejada pelo estudante em questão. Além disso, aceitar/pedir ajuda virtual ou não na remoção de eventuais bugs configura uma violação à esse item.

Não compartilhe seu código com outros estudantes.

Não entregue seu código para nenhum estudante que possa vir a pedi-lo, e não peça o código de outros estudantes. Semelhantemente, não discuta estratégias algorítmicas que impliquem em códigos semelhantes. Discuta idéias com muito cuidado para não se colocar em situação de risco na avaliação.

Não procure e/ou olhe soluções de outros estudantes ou pessoas, seja na web ou não.

A proibição de olhar eventuais soluções de codificação são muito importantes e pertinentes à qualquer curso de programação. Desenvolver bons enunciados de exercícios, trabalho e provas leva muito tempo. Quando um novo enunciado é criado, problemas aparecem fazendo com que o mesmo seja adequadamente ajustado. Sendo assim, através dos anos os enunciados são reutilizados com algumas alterações que visam a melhoria do processo ensino-aprendizagem. Usar soluções de outros, seja do semestre atual ou não, ou mesmo da web, configura uma prática perigosa, uma vez que entre um semestre e outro os enunciados apresentam modificações entre si, sendo que a solução apresentada em um semestre pode não ser adequada ao próximo. Além disso é uma prática que fere o código de conduta e não será tolerada.

Esteja preparado para explicar qualquer código que você submeteu como seu.

Tome cuidado ao requisitar qualquer tipo de ajuda, pois, como o objetivo de um enunciado é verificar seu desempenho, e uma vez que o professor esteja acompanhando o andamento dos estudantes no semestre, você pode ser requisitado a explicar seu código, com o intuito de esclarecer eventuais diferenças de desempenho apresentado na avaliação e nas aulas. Não pense que qualquer trecho de código, por menor que seja, funcionará mágicamente na sua solução sem que você tenha que entendê-lo satisfatoriamente. Muitos estudantes procuram adotar medidas para disfarçar um código alheio como seu próprio código através da mudança de nomes de variáveis e funções, mera alteração de estruturas de repetição e condicionais, expressões lógicas e comentários. Não perca seu tempo fazendo isso. O professor possui métodos eficazes para detecção de plágio, alguns desses métodos são advindos da experiência no ensino e outros são automatizados.

Por fim, o professor em questão observará os itens levantados no código de conduta com o intuito de detectar violações, e assim aplicará as devidas sanções aos envolvidos.

## Cronograma

As datas são prováveis e podem sofrer alterações frente à imprevistos durante o semestre letivo.

### Parte 1: Fundamentos de Lógica e Programação de Computadores

#### Semana 1 Recepção aos Ingressantes

12/FEV	Apresentação da Disciplina	Este documento Notas de Aula 0
--------	----------------------------	-----------------------------------

#### Semana 2 Introdução à Computação

19/FEV	Olá mundo	Notas de Aula 1 Prática em Laboratório
--------	-----------	---

#### Semana 3 Algoritmos

26/FEV	Meus primeiros algoritmos	Atividade prática Prova de Proficiência aos alunos em RE e DP
--------	---------------------------	--

#### Semana 4 Linguagens de Programação

05/MAR		Notas de Aula 3 Hands On ASCENCIO - Cap.2
--------	--	---

#### Semana 5 Fluxogramas, Comandos Básicos e Variáveis

12/MAR	Introdução ao Teste de Mesa	Notas de Aula 4 FORBELLONE - Cap.2.pgs[-18,25,27]; ASCENCIO - Cap.3
	Atividade Prática	

#### Semana 6 Operadores e Estruturas Condicionais

19/MAR	Atividades Práticas	Notas de Aula 5 ASCENCIO - Cap.4; FORBELLONE - Cap.2, Cap.3.pgs[-47]
	Exercícios Guiados	

#### Semana 7 Exercícios (Reposição on line)

19/MAR		Notas de Aula 5
--------	--	-----------------

#### Semana 8 Estruturas de Repetição

26/MAR	Exercícios Guiados	Notas de Aula 6 ASCENCIO - Cap.5; FORBELLONE - Cap.3.pgs[48-]
	Teste de Mesa	

Mega Lista de Exercícios

#### Semana 9 Estruturas de Repetição

02/ABR	Testes de Mesa	Exercícios de Fixação
	Exercícios Guiados	

## Atividades Práticas

---

Semana 10 Avaliação

Ref. *Av<sub>1</sub>*

09/ABR

### Parte 2: Estruturas e Técnicas Complementares

---

Semana 11 Vetores e Matrizes

Notas de Aula 7

16/ABR

ASCENCIO - Cap.6, Cap.7; FORBELLONE - Cap.4.pgs[-83]

Retorno da Avaliação

Ref. *Av<sub>1</sub>*

---

Semana 12 Atividades Práticas

23/ABR Exercícios Guiados

Semana 13 Modularização (Sub-Rotinas)

Notas de Aula 8

30/ABR

FORBELLONE - Cap.6; ASCENCIO - Cap.8

Semana 14 Modularização (Sub-Rotinas)

Notas de Aula 8

07/MAI

FORBELLONE - Cap.6; ASCENCIO - Cap.8

---

Semana 15 Atividades Práticas

14/MAI Exercícios Guiados

Semana 16 Atividades Práticas

21/MAI Exercícios Guiados

Semana 17 Avaliação

Ref. *Av<sub>2</sub>*

28/MAI

Semana 18 Avaliação

Ref. *Av<sub>3</sub>*

11/JUN

Semana 19 Sem atividades

Ref. *SUB*

18/JUN

Semana 20 Avaliação Substitutiva

25/JUN