

Introdução

Algoritmos e Programação de Computadores – ABI/LFI/TAI



Prof. Daniel Saad Nogueira
Nunes

IFB – Instituto Federal de Brasília,
Campus Taguatinga



Sumário

- 1 Introdução
- 2 Ferramentas
- 3 Metodologia
- 4 Considerações
- 5 Bibliografia



Sumário

1 Introdução



Introdução

- Os seres humanos sempre foram fascinados por automatizarem ou facilitarem as suas tarefas.
- O conceito de computador é bem anterior ao século XX.
- Algoritmos precisam ser desenvolvidos para que possam operar nos computadores.



Introdução: Algoritmo

Algoritmo

Um algoritmo é uma sequência finita de passos simples, e bem definidos, capaz de realizar uma determinada tarefa.



Introdução: Algoritmo

Exemplos de Algoritmos

- Algoritmos de soma/subtração/multiplicação de dois números.
- Algoritmo de fatoração de um inteiro em números primos.
- Algoritmo de Euclides para encontrar o MDC.



Algoritmo de Multiplicação

$$\begin{array}{r} 35 \\ \times 12 \\ \hline 70 \\ + 350 \\ \hline 420 \end{array}$$



APC

- Na disciplina de **APC** aprenderemos a desenvolver algoritmos para resolver problemas computacionais.
- Os algoritmos serão implementados utilizando a linguagem de programação C.



O que não é APC

Neste curso:

- Não aprenderemos a manipular o Sistema Operacional.
- Não aprenderemos a utilizar e configurar programas.
- Utilize o Google! É importante ser safo, tecnicamente falando, nestas questões.



Sumário

2 Ferramentas



Ferramentas

- Você precisará de ter acesso a um computador, com um editor de texto e compilador para a linguagem C instalado.
- O editor pode ser um editor simples, como o bloco de notas ou o nano, ou mais complexos, como o Sublime ou o VisualCode.
- O compilador utilizado será o GCC (GNU Compiler Collection) para a linguagem C.
- https://en.wikipedia.org/wiki/List_of_text_editors.
- Já falei para usarem o Google?
- Opcionalmente vocês podem utilizar o depurador GDB para procurar *bugs* nos códigos.



Ferramentas

Replit

- Existe uma ferramenta gratuita e totalmente online chamada replit que possibilita a escrita, compilação e execução de códigos na linguagem C.
- <https://replit.com/>.
- Vem com a ferramenta GDB integrada.
- Vocês podem utilizá-la, mas também considero muito importante saber configurar o seu ambiente de desenvolvimento.



Sumário

3 Metodologia



Metodologia

- As aulas estão organizadas de maneira assíncrona e dispostas através de vídeos.
- Pontos de encontro síncrono para discussão de exercícios.
- Avaliação será individual e dividida em seis projetos de programação. A média ponderada dos projetos corresponde à nota final da disciplina.



$$N_f = \frac{P_1 + P_2 + P_3 + 2P_4 + 2P_5 + 2P_6}{12}$$

- Presença: computada de acordo com a entrega dos projetos.
- Para aprovação: $N_f \geq 6$ e presença superior ou igual à 75%.



Metodologia

O que é preciso para ir bem na disciplina:

- Muita prática: façam os exercícios das listas.
- Não deixe acumular conteúdo. Siga o cronograma da disciplina fielmente.
- Reserve um tempo para a execução dos projetos. Não é interessante deixá-los para véspera.



Sumário

4 Considerações



Considerações Finais

- Site da disciplina: <https://danielsaad.com/algoritmos-e-programacao-de-computadores-tai>
- Horário de atendimento: Segundas-feiras das 18h às 20h.
Agendamento via Calendly (link disponível no site da disciplina).
- Sala de aula virtual: link disponível no site da disciplina.



Sumário

5 Bibliografia



Bibliografia

Ana Fernanda Gomes Ascencio and Edilene Aparecida Veneruchi de Campos, *Fundamentos da programação de computadores*, Pearson Education, 2008.

Harvey Deitel and Paul Deitel, *Como programar em C*, LTC, 1999.

HF Eberspacher and ALV Forbellone, *Lógica de programação: A construção de algoritmos e estruturas de dados*, Pearson Prentice Hall, 2000.

Dennis M Ritchie, Brian W Kernighan, and Michael E Lesk, *The c programming language*, Prentice Hall Englewood Cliffs, 1988.

Jean-Paul Tremblay, *Ciência dos computadores: uma abordagem algorítmica*, McGraw-Hill, 1983.