

Plano de Ensino – Algoritmos e Programação de Computadores

Instituto Federal de Educação, Ciência e Tecnologia de Brasília

Campus Taguatinga



1 Identificação da Disciplina

- Nome da Disciplina: Algoritmos e Programação de Computadores;
- Curso: Tecnologia em Automação Industrial;
- Pré-requisitos: disciplina sem pré-requisitos;
- Carga Horária: 72 h/a;
- Período: 2021/1;
- Professor: Daniel Saad Nogueira Nunes;
- Horário de atendimento: segundas-feiras das 18h às 20h.

2 Bases Tecnológicas (Ementa)

Conceito e desenvolvimento de algoritmos. Tipos de dados. Operações de entrada e saída. Estruturas fundamentais: sequência, decisão e repetição. Vetores e matrizes. Funções. Implementação de algoritmos usando uma linguagem de programação.

3 Objetivos e Competências

- Desenvolver algoritmos utilizando estruturas da programação procedural na linguagem C.
- Adquirir competências na modelagem de um problema em termos computacionais e na sua solução através de um algoritmo escrito em uma linguagem de programação alto-nível.
- Familiarizar com os conceitos básicos da Ciência da Computação.

4 Habilidades Esperadas

- Ser capaz de desenvolver soluções computacionais utilizando uma linguagem de programação alto-nível.

5 Conteúdo Programático

1. Introdução à disciplina;
2. História da Computação;
3. Aritmética Computacional;
4. Atribuição, tipos primitivos, operadores lógicos e aritméticos;
5. Operações de Entrada e Saída;
6. Estruturas de decisão;
7. Estruturas de repetição;
8. Funções;
9. Vetores;
10. Strings;
11. Matrizes.

6 Metodologias de Ensino

Metodologia híbrida: aulas expositivas e aprendizagem baseada em projetos.

7 Recursos de Ensino

Os recursos de ensino baseiam-se, mas não são limitados em:

- Computador;
- Internet;
- Quadro branco, pincel e apagador;
- Projetor multimídia;
- Visitas técnicas e participação em eventos;
- Grupo de discussão restrito da disciplina.

8 Avaliação

A nota da disciplina consiste em três provas.

A nota final é calculada como:

$$N_f = \frac{P_1 + P_2 + P_3 + 2P_4 + 2P_5 + 2P_6}{9}$$

, em que P_i consiste na nota do i -ésimo projeto.

O aluno é considerado **aprovado** se, e somente se, obtiver $N_f \geq 6.0$ e presença $\geq 75\%$. A presença é computada de acordo com a entrega das atividades assíncronas.

9 Observações

Será atribuída nota **ZERO** a qualquer avaliação que incida em plágio.

10 Cronograma

Segue abaixo o planejamento de atividades da disciplina (sujeito à alterações):

Bibliografia

- [AB09] Sanjeev Arora and Boaz Barak, *Computational complexity - A modern approach*, Cambridge University Press, 2009.
- [CLRS09] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, *Introduction to algorithms (3. ed.)*, MIT Press, 2009.

Semana	Conteúdo	Total de Horas
18/mai	Introdução à disciplina (S) e História da Computação (A)	4
25/mai	Aritmética Computacional (A)	4
01/jun	Variáveis, Atribuição, tipos primitivos de dados e Operadores (A)	4
08/jun	Entrada e Saída (A)	4
15/jun	Estruturas de Decisão e Projeto 1 (A)	4
22/jun	Estruturas de Decisão e Projeto 1 (A)	4
29/jun	Estruturas de Repetição e Projeto 2 (A)	4
06/jul	Estruturas de Repetição e Projeto 2 (A)	4
13/jul	Estruturas de Repetição e Projeto 2 (A)	4
20/jul	Funções (A) e Projeto 3 (A)	4
27/jul	Funções (A) e Projeto 3 (A)	4
03/ago	Vetores (A) e Projeto 4 (A)	4
10/ago	Vetores (A) e Projeto 4 (A)	4
17/ago	Strings (A) e Projeto 5 (A)	4
24/ago	Strings (A) e Projeto 5 (A)	4
31/ago	Matrizes (A) e Projeto 6 (A)	4
07/set	Matrizes (A) e Projeto 6 (A)	4
14/set	Projeto 6 (A) e Encerramento da Disciplina (S)	4

Total 72
 Legenda: (S) Síncrono
 (A) Assíncrono

- [Knu68] Donald E. Knuth, *The art of computer programming, volume I: fundamental algorithms*, Addison-Wesley, 1968.
- [Knu69] ———, *The art of computer programming, volume II: seminumerical algorithms*, Addison-Wesley, 1969.
- [Knu73] ———, *The art of computer programming, volume III: sorting and searching*, Addison-Wesley, 1973.
- [Man89] Udi Manber, *Introduction to algorithms - a creative approach*, Addison-Wesley, 1989.
- [Pap07] Christos H. Papadimitriou, *Computational complexity*, Academic Internet Publ., 2007.
- [Ski08] Steven Skiena, *The algorithm design manual (2. ed.)*, Springer, 2008.