

Plano de Ensino – Análise de Algoritmos

Instituto Federal de Educação, Ciência e Tecnologia de Brasília

Câmpus Taguatinga



1 Identificação da Disciplina

- Nome da Disciplina: Análise de Algoritmos;
- Curso: Bacharelado em Ciência da Computação;
- Pré-requisitos: Algoritmos e Programação de Computadores;
- Carga Horária: 72 h/a.
- Período: 2021/1;
- Professor: Daniel Saad Nogueira Nunes.

2 Bases Tecnológicas (Ementa)

Modelos computacionais. Cotas inferiores e superiores. Medidas de eficiência de algoritmos. Técnicas de projeto e análise de algoritmos. Algoritmos de ordenação e busca. Redutibilidade. Complexidade computacional. Classes de problemas. Problemas NP-completos. Tratamento de problemas NP- difíceis.

3 Objetivos e Competências

- Estudar métodos de análise de algoritmos e relações de recorrência.
- Verificar paradigmas de projeto de algoritmos.
- Detectar a dificuldade inerente de problemas.

4 Habilidades Esperadas

- Analisar as soluções propostas quanto aos recursos de tempo/espço em termos assintóticos.
- Dominar os paradigmas de divisão e conquista, algoritmos gulosos e programação dinâmica para projeto de algoritmos.
- Identificar a intratabilidade de problemas.

5 Conteúdo Programático

1. Introdução à disciplina.
2. Conceitos preliminares.
3. Notação assintótica.
4. Relações de recorrência.
5. Projeto por indução.
6. Algoritmos gulosos.
7. Programação dinâmica.
8. Casamento de padrões.
9. Compressão de dados.
10. Classes de complexidade P e NP.
11. Problemas NP-Completo.
12. Redução de problemas.
13. Tratamento de problemas difíceis.

6 Metodologias de Ensino

A metodologia de ensino tradicional é adotada na disciplina, mas com atenção de procurar instigar a participação dos alunos.

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:

$$N_f = \frac{N_1 + N_2 + N_3}{3}$$

Onde N_1, N_2 e N_3 , correspondem as três maiores notas considerando três provas.

O aluno é considerado **aprovado** se, e somente se, obtiver $N_f \geq 6.0$ e presença $\geq 75\%$.

9 Observações

Será atribuída nota **ZERO** a qualquer avaliação onde for detectado plágio tanto para quem forneceu, para quem plagiou.

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.

Tabela 1: Cronograma.

Semana do dia	Conteúdo	Total de Horas
17/fev	Introdução à disciplina e Análise Assintótica	4
2/mar	Análise Assintótica	4
9/mar	Relações de Recorrência	4
16/mar	Ordenação	4
23/mar	Ordenação	4
30/mar	Exercícios	4
06/abr	Prova 1	4
13/abr	Algoritmos Gulosos	4
20/abr	Projeto por Indução	4
27/abr	Programação Dinâmica	4
04/mai	Programação Dinâmica	4
11/mai	Exercícios	4
18/mai	Prova 2	4
25/jun	As classes P e NP	4
01/jun	Problemas NP-Completos	4
08/jun	Redutibilidade	4
15/jun	Exercícios	4
22/jun	Prova 3	4
29/jun	Encerramento da Disciplina	4
Total		76

- [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.