

Análise de Algoritmos e Estrutura de Dados

Objetivos

- Apresentar noções e conceitos de complexidade de computação;
- Apresentar métodos e conceitos que permitam ao aluno, de maneira confiável, avaliar a qualidade de um algoritmo. A essência destes métodos e conceitos estará focalizada no cálculo de complexidade e prova de corretude de algoritmos;
- Caracterizar técnicas gerais de desenvolvimento de algoritmos que permitam ao aluno melhor projetá-los conforme sua natureza. As técnicas gerais escolhidas a serem estudadas são Divisão e Conquista, Método Guloso e Programação Dinâmica;
- Apresentar noções básicas de Classes de Complexidade, em particular as classes P, NP e NP-Completo.

Ementa da disciplina

Tempo de execução e análise assintótica. Corretude de algoritmos iterativos e recursivos. Algoritmos de ordenação. Recorrências e técnicas de solução de recorrências. Técnicas de projeto de algoritmos: divisão e conquista, método guloso e programação dinâmica. Algoritmos em grafos para árvores geradoras mínimas e caminhos mínimos. Introdução à teoria da complexidade computacional: redução entre problemas e classes P, NP, NP-completo e NP-difícil.

Recomendações

Para facilitar o acompanhamento do curso, é recomendado que você possua:

- conhecimentos de programação (em qualquer linguagem imperativa), com boas noções de algoritmos recursivos,
- familiaridade com estruturas de dados básicas (vetores, listas, pilhas, filas e árvores),
- capacidade para reconhecer argumentos lógicos em uma prova matemática (por indução, contradição, construção),
- familiaridade com linguagem matemática (como quantificadores lógicos, somatórios e manipulação de funções).

Outros materiais de apoio:

- [Livro de Bases Matemáticas](#), dos profs. Armando Caputi e Daniel Miranda, da UFABC.
- [Fundamentos da matemática para computação](#), (videoaulas) do prof. Cláudio Possani, da USP.
- Minhas aulas de [revisão de Matemática Discreta](#).

- Aulas de revisão de indução da profa. Carla Lintzmayer: [definição](#) e [exemplos](#).
- [Projeto de algoritmos \(em C\)](#), do prof. Paulo Feofiloff, da USP.
- [Estruturas de dados \(em C\)](#), do prof. Paulo Feofiloff, da USP.
- [Notas de aula](#) da disciplina de Estruturas de Dados do prof. Rafael Schouery, da Unicamp (introdução à programação em C, recursão, listas, pilhas e filas, árvores).
- [Notas de aula](#) da disciplina de Programação Estruturada da profa. Carla Lintzmayer (introdução à programação em C, recursão, vetores e listas).

Bibliografia e outros materiais

1. [LM] Lintzmayer, C. N.; Mota, G. O.. [Notas de aulas - Análise de algoritmos e estruturas de dados](#) (o conteúdo dessa disciplina está completo no livro, mas sempre existem atualizações - verifique sempre sua versão).
2. [CLRS3] Cormen, T. H.; Leiserson, C. E.; Rivest, R. L.; Stein, C.. *Introduction to Algorithms* . 3rd ed. MIT Press. 2009.
3. [CLRS2] Cormen, T. H.; Leiserson, C. E.; Rivest, R. L.; Stein, C.. *Introduction to Algorithms*. 2nd ed. MIT Press. 2002.
4. [Ma] Manber, U.. *Introduction to Algorithms: A Creative Approach*. 1989

Critérios de avaliação regular

A média final antes da REC (**MF**) será calculada da seguinte forma:

$$MF = 0.5P1 + 0.5P2,$$

onde,

- **P1** e **P2** são as notas da primeira e segunda avaliações, respectivamente.

$$CF = \begin{cases} \mathbf{A}, & \text{se } MF \in [8.5; 10.0] \\ \mathbf{B}, & \text{se } MF \in [7.0; 8.5) \\ \mathbf{C}, & \text{se } MF \in [6.0; 7.0) \\ \mathbf{F}, & \text{se } MF < 5.0 \end{cases}$$

Caso seja verificado ocorrência de fraude acadêmica, o aluno será automaticamente reprovado com F.

Dias, horários e locais das aulas

- Terças-feiras: 16h - 18h (sala S-305-1);
- Quintas-feiras: 14h - 16h (sala S-305-1);

Datas Importantes

- Avaliação 1: 08/08 (quinta-feira)
- Avaliação 2: 17/09 (terça-feira)
- Substitutiva: 19/09 (quinta-feira)

Avaliação substitutiva está disponível apenas para os casos assegurados pelo regulamento da UFABC.

Atendimento

Local: Sala S-305-1

- Quintas-feiras: 16h-18h (Maycon Sambinelli)

Cronograma

Aula	Data	Tópico
1	25/06	Sobre o curso e resolução de exercícios de revisão.
2	27/06	Notação Assintótica
3	02/07	Notação Assintótica
4	04/07	Solução de Recorrências (Substituição)
5	11/07	Solução de Recorrências (Iteração, Árvore de recursão e Mestre)
6	16/07	Introdução à Análise de Algoritmos e Correção de Algoritmos Iterativos
7	18/07	Tempo de Execução de Algoritmos Iterativos
8	23/07	Insertion-Sort: Análise e Correção
9	25/07	Correção e Tempo em Algoritmos Recursivos & Divisão e Conquista & Merge-Sort
10	30/07	Selection-Sort & Heap
11	01/08	Heap & Heapsort. Limitante Inferior da Ordenação
12	06/08	Introdução a Grafos e à Análise de Algoritmos para Grafos
13	08/08	Avaliação 1
14	13/08	Busca em Largura & Distância
15	15/08	Programação Dinâmica
16	22/08	Programação Dinâmica
17	27/08	Algoritmos Gulosos
18	29/08	Algoritmos Gulosos
19	03/09	Redução entre problemas.
20	05/09	Classes P, NP, NP-completo e NP-difícil. Problemas NP-completos.
21	10/09	Classes P, NP, NP-completo e NP-difícil. Problemas NP-completos.
22	12/09	Exemplo Extra: Programação Dinâmica ou Algoritmos Gulosos
23	17/09	Avaliação 2
24	19/09	Substitutiva