

MCTA003-17 -- Análise de Algoritmos -- 2024.2

Jun 3, 2024

Atualizado em 17/06

Expediente

- Professor: [Aritanan Gruber](#)
- Moodle: [AA \(Q24.2D\)](#)
(detalhes de andamento do curso, links úteis, avaliações, notas, etc.)
- Horário: Qua 10–12h e Sex 08–10h na S-214.0
- Atendimento: Sex 10–12h na S-539.2
- Monitoria: (a definir)

Ementa

Conceitos básicos: recorrências, medidas de complexidade (melhor caso, caso médio e pior caso). Técnicas gerais de projeto de algoritmos: divisão e conquista, método guloso e programação dinâmica. Classes de complexidade: P, NP e NP-completude.

Pré-requisitos recomendados

Matemática Discreta, Algoritmos e Estruturas de Dados I

Avaliações e critérios

- listas de exercícios opcionais (não contam para nota, mas é **necessário** fazê-las para ter chance de aprovação)
- duas provas regulares P_1 e P_2 e uma prova substitutiva P_3 *aberta*;
média aritmética das 2 melhores dentre as 3 possibilidades
 - P1 : Qua 31/07 @ 10h
 - P2 : Qua 11/09 @ 10h
 - P3 : Sex 13/09 @ 08h

Nota nominal:

$$N = \frac{1}{2} \max \left\{ \sum_{j \in S} P_j : S \in \binom{[3]}{2} \right\} = \frac{1}{2} \max \{P_1 + P_2, P_1 + P_3, P_2 + P_3\},$$

com $P_i \in [0, 10]$.

Conceito nominal (C_N): reflete o seu desempenho frente ao material apresentado e às avaliações realizadas; obtido pelo encaixe de N em um dos intervalos:

$$-\infty < \mathbf{F} < 5.0 \leq \mathbf{D} \leq 6.0 < \mathbf{C} \leq 7.0 < \mathbf{B} \leq 8.5 < \mathbf{A} < \infty.$$

Recuperação

Caso seu conceito C_N seja **D** ou **F**, você tem direito a uma prova de recuperação P_R . Esta será única e contemplará toda a matéria do quadrimestre. Uma nova nota nominal $\bar{N} = (N + P_R)/2$ será utilizada para gerar um novo conceito (nominal) *final pós-recuperação* \bar{C}_N . Note que seu conceito final pós-recuperação pode ser menor que o pré-recuperação: uma vez feita, a recuperação é parte integrante da sua avaliação.

Bibliografia

Primária

- **[CLRS]** T. Cormen, C. Leiserson, R. Rivest, C. Stein, [Introduction to Algorithms](#), 3rd ed., MIT University Press (2009) [é possível se virar com a 2a ed.]
Versão em Português: [Algoritmos: Teoria e Prática](#), 3a ed., Elsevier (2012)
[Não confunda com a tradução da 2a edição pela Campus (2001), que tem baixa qualidade.]

Secundária

- [AU] A. Ahu, J.D. Ullman, [Foundations of Computer Science \(C edition\)](#), W.H. Freeman (1994)
- [Er] J. Erickson, [Algorithms](#), self-published (2019)
- [Fe1] P. Feofiloff, [Minicurso de Análise de Algoritmos](#), self-published (2009)
- [Fe2] P. Feofiloff, [Aulas de Análise de Algoritmos](#), em html (2021)
- **[KT]** J. Kleinberg, E. Tardos, [Algorithm Design](#), Pearson Education, Inc. (2006)

Requisitos

- [LLM] E. Lehman, F. Leighton, A. Meyer, [Mathematics for Computer Science](#), MIT CC3.0 (2018)
- **[Fle]** M.M. Fleck, [Building Blocks for Theoretical Computer Science](#)

- [DPV] S. Dasgupta, C. Papadimitriou e U. Vazirani, [Algorithms](#), McGraw-Hill (2006)
Versão em Português: *Algoritmos*, AMGH (2009)

Lista de livros no [assunto](#) curada por [Paulo Feofiloff](#).

Lista de tópicos por semana (tentativa)

Detalhes de cada tópico (coberto nas aulas) serão atualizados no Moodle ao longo do quadrimestre.

Aulas	Datas	Tópicos
A01	26/06	Notação Assintótica 1. InsertionSort
A02	28/06	Notação Assintótica 2. Crescimento de funções
A03	03/07	Divisão e Conquista 1: MergeSort e recorrências
A04	05/07	Divisão e Conquista 2: QuickSort (melhor e pior casos)
A05	10/07	Análise Probabilística: QuickSort (caso médio)
A06	12/07	QuickSort e Select aleatorizados. Cota inferior para ordenação
A07	17/07	Ordenação em tempo linear: Counting, Radix e Bucket Sorts
A08	19/07	Divisão e Conquista 3: Karatsuba e Strassen
A09	24/07	Programação Dinâmica 1: Fibonacci, Cortes de hastes
A10	26/07	Programação Dinâmica 2: Produto de cadeias de matrizes
A11	31/07	Prova P1
A12	02/08	Programação Dinâmica 3: Subsequência comum mais longa
A13	07/08	Programação Dinâmica 4: Árvore binária de busca ótima
A14	09/08	Programação Dinâmica 5: Problemas das mochilas binária e inteira
A15	14/08	Algoritmos Gulosos 1: Intervalos disjuntos, coloração de intervalos, mochila fracionária
A16	16/08	Algoritmos Gulosos 2: Código de Huffman, um problema de escalonamento
A17	21/08	Algoritmos Gulosos 4: Mais escalonamentos, Matróides e o método guloso
A18	23/08	Algoritmos Gulosos 5: Árvores geradoras mínimas: Kruskal e Prim
A19	28/08	Caminhos mínimos em grafos: Dijkstra (guloso) e Floyd-Warshall (prog. din.)
A20	30/08	Análise Amortizada: Tabelas dinâmicas, Union-Find
A21	04/09	Complexidade Computacional I
A22	06/09	Complexidade Computacional II
A23	11/09	Prova P2
A24	13/09	Prova P3 (substitutiva)

Estudando para esta disciplina

A natureza do tópico, o posicionamento do curso na grade, e a lista de pré-requisitos indicam que esta é uma disciplina de nível intermediário; e será tratada como tal. Você deve assistir às aulas, estudar a bibliografia indicada, e dedicar-se às listas de exercícios.

Caso seus pré-requisitos não estejam tão sólidos quanto desejável (falta de familiaridade com formalismo matemático e raciocínio algorítmico, atitude passiva com relação ao aprendizado, tempo dedicado insuficiente, etc.), será possível fazer confusões e sentir-se perdido no início.

Alguns procedimentos que costumam funcionar em cursos introdutórios para mitigar os motivos relacionados também costumam funcionar por aqui:

- Refaça os exemplos e re-prove os resultados fornecidos em sala de aula.
- Preste atenção aos processos de solução (aprenda-os!) e não foque somente nos resultados finais.
- Assista ativamente às aulas; resolva os exercícios nelas propostos e os contidos nas listas.
- Estude a bibliografia indicada, monte grupos de estudo, e faça um bom uso dos horários de atendimento.
- Tenha sempre em mente que aprendizado é uma tarefa *ativa*; não fique somente assistindo. “Ouvir” às aulas e “ler” os livros tem pouco ou nenhum efeito neste curso – e em qualquer disciplina matemática/algorítmica que o valha.
- Se ainda assim, sentir-se perdido, repita os passos acima. Mais cedo ou mais tarde, eles convergirão à compreensão.

Note que você não será convidado a regurgitar respostas fornecidas em aula ou presente nos livros. As questões em listas e provas testarão sua capacidade de entender os problemas e apresentar uma solução para eles; às vezes, serão uma adaptação simples ou uma extensão direta do que foi visto, outras, será necessário relacionar dois ou mais métodos ou conceitos apresentados, e outras ainda, irão requerer análise e raciocínio mais profundo (o que leva tempo, então não deixe nada para a última hora!).

Tenha em mente: além do escrito acima, para aproveitar bem este curso, você deve ler os slides e familiarizar-se com o material na leitura sugerida correspondentes antes da aula, e estudá-los com afinco depois.

Integridade acadêmica e transgressões

O Artigo 25 do [Código de Ética da UFABC](#) estabelece, à página 23: “Quanto aos trabalhos acadêmicos, é eticamente inaceitável que os discentes:

- I - fraudem avaliações;
- II - fabriquem ou falsifiquem dados;
- III - plajeiem ou não creditem devidamente autoria;
- IV - aceitem autoria de material acadêmico sem participação na produção;
- V - vendam ou cedam autoria de material acadêmico próprio a pessoas que não participaram da produção.”

Trabalhos (listas, provas, programas) suspeitos de cópia ou de outra representação fraudulenta acarretarão aos envolvidos conceitos **F** (falha) no curso. A atividade será reportada à [Comissão Disciplinar Discente](#) da universidade para que sejam tomadas todas as providências disciplinares cabíveis.

LLMs: ChatGPT e similares

Antes de mais nada, representam um grande avanço da inteligência artificial generativa. São resultados magníficos! Mas, a função objetivo que está sendo otimizada é exatamente a produção de textos da forma como os humanos escrevem. Não há mecanismos de “raciocínio” subjacentes. Fornecem respostas certas ou erradas com o mesmo grau de certeza e confiança! E, apesar dos recentes avanços do ChatGPT-4o e do Gemini Pro, costumam produzir respostas a questões matemáticas (não elementares) com inúmeros erros! Além disto, soluções entregues que tenham sido produzidas por eles enquadram-se no Artigo 25 acima.

Para pensar ao longo do curso: *Do que adianta as máquinas aprenderem e os alunos não?*



Aritanan Gruber

Assistant Professor

“See, if y’all haven’t the same feeling for *this*, I really don’t give a damn. If you ain’t feeling it, then dammit *this* ain’t for you!”
(desconheço a autoria; agradeço a indicação)

[✉](#) [G](#) [R](#) [ID](#) [in](#) [CV](#)

