



Plano de Ensino

Universidade Federal do Espírito Santo

Campus de São Mateus

Curso: Ciência da Computação - São Mateus

Departamento Responsável: Departamento de Computação e Eletrônica

Data de Aprovação (Art. nº 91): 27/08/2025

DOCENTE PRINCIPAL : LUCIANA LEE

Matrícula: 2509987

Qualificação / link para o Currículo Lattes: <http://lattes.cnpq.br/2240966624034107>

Disciplina: PROJETO E ANÁLISE DE ALGORITMOS

Código: DCE16377

Período: 2025 / 2

Turma: 3704.1

Pré-requisito:

Carga Horária Semestral: 60

Disciplina: DCE16358 - MATEMÁTICA DISCRETA

Disciplina: DCE16376 - ESTRUTURA DE DADOS II

Distribuição da Carga Horária Semestral

Créditos: 4	Teórica	Exercício	Laboratório	Extensão
	60	0	0	

Ementa:

Complexidade de tempo e espaço de algoritmos. Análise assintótica. Introdução à classes de complexidade. Relação de recorrência. Análise de pior caso e análise probabilística. Cota inferior de ordenação. Paradigmas de projeto de algoritmos e métodos de análise. Estudo de casos. Introdução à teoria da complexidade computacional. As classes P, NP, e NP-completo. Redução entre problemas. Algoritmos probabilísticos.

Objetivos Específicos:

Consolidar conceitos de análise da correção e do desempenho de algoritmos. [Familiaridade]

Explicar o significado de "melhor", "esperado" e "pior" caso de um algoritmo. [Familiaridade]

Desenvolver a habilidade de projetar algoritmos e estimar seu desempenho. [Familiaridade]

Descrever o uso de relações de recorrência para determinar a complexidade de tempo de algoritmos definidos recursivamente. [Uso]

Resolver relações de recorrência elementares usando, por exemplo, o Teorema Mestre. [Uso]

Analisar a complexidade dos principais algoritmos de ordenação (simples e eficientes) e apresentar uma cota inferior de tais algoritmos. [Familiaridade]

Descrever os conceitos dos paradigmas de projeto de algoritmos: força bruta, programação dinâmica, divisão e conquista e algoritmo guloso. [Familiaridade]

Identificar exemplos aplicáveis a cada um dos paradigmas de projeto de algoritmos. [Avaliação]

Apresentar as características que diferenciam os problemas que podem ser resolvidos por algoritmos gulosos dos que devem ser resolvidos usando programação dinâmica. [Familiaridade]

Apresentar um estudo de caso da análise de alguns problemas clássicos como: algoritmo quicksort aleatorizado, tabelas de hashing, problema da mochila, multiplicação de inteiros (algoritmo de Karatsuba) e matrizes (algoritmo de Strassen), árvores geradoras mínimas de grafos (algoritmos de Prim e Kruskal), entre outros. [Uso]

Introduzir noções da teoria da complexidade computacional, incluindo as classes P, NP, e NP-completo, e redução entre problemas. [Familiaridade]

Conteúdo Programático:

- 1- Elementos de análise assintótica - notação O, Omega e Theta.
- 2- Solução de recorrências.
- 3- Análise da correção e desempenho de algoritmos iterativos.
- 4- Análise da correção e desempenho de algoritmos recursivos.
- 5- Análise de pior caso e análise probabilística (caso médio).
- 6- Algoritmos de busca e ordenação.
- 7- Algoritmos de programação dinâmica.
- 8- Algoritmos gulosos.
- 9- Algoritmos para problemas em grafos.
- 10- Introdução à teoria da complexidade: problemas completos em NP

Metodologia:

Aulas expositivas em sala de aula e listas de exercícios.

Critérios / Processo de avaliação da Aprendizagem :

A avaliação será composta de duas provas. A média parcial (MP) é dada pela média aritmética das notas das provas. Caso o aluno não atinja a média 7,0, terá que fazer a Prova Final (PF) e a média final (MF) é dada por:
 $MF = (MP+PF)/2$.

Bibliografia básica:

CORMEN, Thomas H.; LEISERSON, Charles E.; RIVEST, Ronald L.; STEIN, Clifford. Algoritmos. Teoria e Prática. 3a ed. Rio de Janeiro: Rio de Janeiro: Elsevier - Campus, 2012.
KLEINBERG, Jon; TARDOS, Éva . Algorithm Design. 1a ed. Pearson, 2005.
ZIVIANI, Nivio. Projeto de Algoritmos com Implementações em Java e C++. 1a ed. Cengage Learning, 2006.

Bibliografia complementar:

CORMEN, Thomas H.; LEISERSON, Charles E.; RIVEST, Ronald L.; STEIN, Clifford. Introduction to algorithms. 3rd ed. Cambridge, Mass.: The MIT Press; New York: McGraw-Hill, 2009. xix,1292 p.
PAPADIMITRIOU, Christos H.; DASGUPTA, Sanjoy; VAZIRANI, Umesh. Algorithms. 1st ed. McGraw-Hill, 2007.
KNUTH, Donald E. The Art of Computer Programming, vols. 1 e 3. 3th ed. Addison-Wesley, 1997.
SKIENA, Steven S. The Algorithm Design Manual. 2nd ed. Springer, 2011.
GOODRICH, Michael T.; TAMASSIA, Roberto. Algorithm Design and Applications. 1st ed. Wiley, 2014.

Cronograma:

Observação:



UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO

PROTOCOLO DE ASSINATURA



O documento acima foi assinado digitalmente com senha eletrônica através do Protocolo Web, conforme Portaria UFES nº 1.269 de 30/08/2018, por
LUCIANA LEE - SIAPE 2509987
Departamento de Computação e Eletrônica - DCE/CEUNES
Em 26/08/2025 às 12:38

Para verificar as assinaturas e visualizar o documento original acesse o link: <https://api-lepisma.prod.ukf.ufes.br/arquivos-assinados/1191302?tipoArquivo=O>