

PLANO DE ENSINO

DEPARTAMENTO: Ciência da Computação

DISCIPLINA: Complexidade de Algoritmos

SIGLA: CAL0001

CARGA HORÁRIA TOTAL: 72

TEORIA: 36

PRÁTICA: 36

CURSO(S): Bacharelado em Ciência da Computação

SEMESTRE/ANO: 2/2018

PRÉ-REQUISITOS: TEG0001

OBJETIVO GERAL DA DISCIPLINA: Analisar a complexidade de tempo e espaço de algoritmos. Identificar o melhor caso, o pior caso e o caso médio de execução de algoritmos. Identificar problemas tratáveis e intratáveis.

EMENTA: Crescimento assintótico de funções. Somatórios. Análise de complexidade de algoritmos. Algoritmos iterativos e recursivos. Divisão e conquista. Algoritmos gulosos. Programação Dinâmica. Problemas tratáveis e intratáveis. Classes de problemas: P, NP, NP-Completo e NP-Difícil. Aproximações e Heurísticas.

OBJETIVOS ESPECÍFICOS:

Capacitar o aluno a analisar a complexidade de tempo e espaço de algoritmos e ser capaz de identificar problemas considerados intratáveis.

C.H.	CONTEÚDOS PROGRAMATICOS
14 h/a	Apresentação do planejamento e da ementa da disciplina. Introdução à disciplina. Conceitos Básicos de Complexidade: - Notação O grande - Ordens de complexidade - Análise de complexidade com uma variável - Análise de complexidade de tempo de algoritmos recursivos - Somatórios - Complexidade de Espaço - Notações assintóticas (Ω e Θ) - Teorema Mestre (Master)
6 h/a	Análise de Algoritmos de Ordenação: Merge Sort, Quick Sort, Heap Sort, Counting Sort e Bucket Sort
8 h/a	Análise de Complexidade de Estruturas de Dados Elementares Tabelas Hash Análise de Complexidade com múltiplas variáveis
12 h/a	Análise de Complexidade de operações elementares com inteiros de 'n' bits Números primos, aritmética modular e algoritmo de criptografia RSA
18 h/a	Abordagens para Resolução de Problemas: - Indução matemática - Divisão e conquista - Algoritmos gulosos - Algoritmos de tentativa e erro

	<ul style="list-style-type: none"> - Programação dinâmica - Algoritmos de aproximação
14 h/a	<p>Conceitos de Teoria da Computação vinculados com complexidade:</p> <ul style="list-style-type: none"> - Problemas tratáveis e intratáveis; - Classes de problemas: P, NP, NP-Completo e NP-Difícil; - Redução de problemas; - Problemas NP-Completo: SAT, 3-CNF-SAT, Clique, Cobertura de Vértices, Ciclo Hamiltoniano, Caixeiro Viajante, Subset-Sum - Algoritmos pseudo-polinomiais

METODOLOGIA PROPOSTA:

Aulas expositivas acompanhadas de trabalhos práticos relacionados aos conteúdos apresentados na aula. Aulas práticas em laboratório que objetivam a implementação ou pesquisa dos conceitos apresentados nas aulas teóricas. Listas de exercícios para auxiliar na fixação do conteúdo apresentado. Seminários para proporcionar a busca de forma autônoma pelo conhecimento. Provas teóricas para avaliar o conteúdo conceitual aprendido. Trabalhos para avaliar a capacidade do uso dos conceitos aprendidos. Até 20% do conteúdo programático poderá ser ministrado na forma de ensino a distância.

AValiação:

Do desempenho do aluno:

Os alunos serão avaliados com base no seu desempenho nas provas, trabalhos e listas de exercícios, sendo que ao final do semestre o aluno deverá ter comparecimento mínimo de 75% às aulas e desempenho mínimo de 70% nas avaliações. O grau de desempenho do aluno será avaliado com base nos seguintes critérios:

- a) Provas: 55%
 - a.1) Prova 1 (20%)
 - a.2) Prova 2 (20%)
 - a.3) Prova 3 (15%)
- b) Trabalhos: 45%
 - b.1) Comparação entre algoritmos de ordenação (10%)
 - b.2) Trabalho comparativo entre busca sequencial, binária e hash (10%)
 - b.3) Pesquisa e apresentação de complexidade de algoritmo (10%)
 - b.4) Implementação e análise de algoritmos para o problema do caixeiro-viajante (15%)

BIBLIOGRAFIA (GERAL) OU DE USO DA DISCIPLINA:

Básica:

CORMEN, Thomas H. Algoritmos: teoria e prática. 3. ed. Rio de Janeiro: Elsevier, 2012.
 CORMEN, Thomas H. Desmistificando algoritmos. Rio de Janeiro: Elsevier, 2014.
 DASGUPTA, Sanjoy; PAPADIMITRIOU, Christos H; VAZIRANI, Umesh Virkumar. Algoritmos. São Paulo: McGraw-Hill, 2009

Complementar:

AHO, A.V; HOPCROFT, J.E.; ULLMAN, J.D.; Data structures and algorithms. Reading, MA: Addison Wesley, 1987.
 KNUTH, D.E. The art of computer programming - Fundamental Algorithms. 3rd ed. Massachusetts: Addison Wesley, 2008.
 LEVITIN, A. Introduction to the design & analysis of algorithms. 2nd ed. Pearson/Addison Wesley, 2007.
 TOSCANI, L.V.; VELOSO, Paulo A. S. Complexidade de algoritmos. 3.ed. Porto Alegre: Bookman, 2012.
 ZIVIANI, N. Projeto de algoritmos: com implementações em Pascal e C. 3. ed. rev. e ampl. São Paulo: Pioneira Thomson Learning, 2011.