

MINISTÉRIO DA EDUCAÇÃO

CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA CELSO SUCKOW DA FONSECA CAMPUS PETRÓPOLIS

CURSO SUPERIOR DE ENGENHARIA DE COMPUTAÇÃO

DEPARTAMENTO		PLANO DE CURSO DA DISCIPLINA			
Engenharia de Computação		Análse de Algoritmos			
CÓDIGO	PERÍODO	ANO	SEMESTRE	PRÉ-REQUISITOS	
GCOM6036PE	4	2016	1		
CRÉDITOS	AULAS/SEMANA			TOTAL DE AULAS NO SEMESTRE	Algoritmos e Estruturas de Dados II
4	TEÓRICA	PRÁTICA	ESTÁGIO	72	
	4	0	0		

EMENTA

1. Conceitos Básicos: motivação e solução de problemas, critérios de análise, correção e eficiência;
2. Análise de recorrência;
3. Teorema Mestre;
4. Análise de Algoritmos: tempo de processamento e operações elementares, complexidade de caso médio e pior caso, algoritmos polinomiais, comparação de algoritmos, algoritmos recursivos e algoritmos pseudo-polinomiais;
5. Programação Dinâmica;
6. Algoritmos Gulosos;
7. Teoria de Complexidade: problemas de decisão, transformações polinomiais, classe P, algoritmos não-determinísticos, class NP, Co-NP e NP-Difícil, problemas NP-Completo;

BIBLIOGRAFIA

Bibliografia Básica:

- SZWARCFITER, J.L.; MARKENZON, L. Estruturas de Dados e seus Algoritmos, LTC, 3ª edição, 2010.
- CORMEN, T.H. et al. Algoritmos: Teoria e Prática. Rio De Janeiro: Campus, 2012.
- CELES, W.; CERQUEIRA, R.; RANGEL, J.L. Introdução a Estruturas de Dados - com Técnicas de Programação em C. Rio de Janeiro: Elsevier (Campus), 2004.

Bibliografia Complementar:

- WIRTH, N. Algoritmos e estruturas de dados. Rio de Janeiro: LTC, c1999.
- BOAVENTURA NETTO, P.O. Grafos: teoria, modelos, algoritmos. 5^a edição revista e ampliada. São Paulo: E. Blucher, 2012.
- ZIVIANI, N. Projeto de algoritmos: com implementações em Java e C++. São Paulo: Thomson, 2007.
- JOHNSONBAUGH, R. Discrete mathematics. New Delhi: Pearson, c2014.
- KOFFMAN, E.B.; WALFGANG, P.A.T. Objetos, abstração, estruturas de dados e projeto usando C++. Rio de Janeiro: LTC, 2008.

OBJETIVOS GERAIS

O principal objetivo desta disciplina prende-se com aprender sobre análise algorítmica. Esta aprendizagem requer que o aluno seja capaz de:

- Definir algoritmos iterativos e recursivos;
- Analisar correção de algoritmos;
- Analisar os tempos de execução dos algoritmos (Notações :Big-O, Little-O, Big-Omega, Little-Omega e Teta);
- Analisar Recorrências
- Aplicar Teorema Mestre;
- Desenhar árvores de decisão;
- Algoritmos de ordenação linear;
 - Tempo linear;
 - Tempo polinomial;
 - Tempo quasi-linear;
- Funções de Hashing;
- Programação Dinâmica:
 - Problema dos números de Fibonacci;
 - Problema dos caminhos mais curtos;
 - Problema de justificação de Textos (sufixos)
 - Problema do Blackjack (sufixos)
 - Problema de Parentização (substrings)
 - Problema da distância de edição (sufixos);
 - Problema da Mochila (sufixos) – tempo pseudo-polinomial;
 - Problema das notas musicais;
 - Problema do Tetris;
- Programação Gulosa:
 - Propriedade Gulosa;
 - Minimum Spanning Trees;
 - Algoritmo de Prim;
 - Single-source shortest paths;
 - Algoritmo de Dijkstra;
- NP-Completeness:
 - Classe P, NP, NP-Hard, NP-Complete;
 - Problema: Super Mario Bros;
 - Problema: 3D Matching;
 - Problema: Subset sum problem;

- Problema: Partition;
- Problema: Rectangle packing
- Problema: Jigsaw Puzzles;

METODOLOGIA

- Aulas teóricas: onde são leccionados os conceitos fundamentais da disciplina e realizados exercícios com o objetivo de facilitar o aprendizado. Os alunos são avaliados no decorrer das aulas teóricas em função da sua participação.
- O ênfase das aulas é mostrar conceitos teóricos e exemplificar aplicações destes para desmistificar a sua complexidade. Um dos objetivos principais é o de incentivar a iniciativa pessoal da parte dos alunos na resolução de problemas.

CRITÉRIO DE AVALIAÇÃO

Critério de Avaliação Principal:

- Duas provas durante o semestre (80% da nota)
- Participação na aula (20% da nota)
- Aprovação neste critério requer que o aluno alcance uma média maior ou igual a 7,0 (sete) pontos.

Critério de Avaliação Secundário:

- Para alunos com nota maior ou igual a 3,0 (três) pontos e menor que 7,0 (sete) pontos na avaliação principal;
- Realização de uma prova final;
- Aprovação neste critério requer que o aluno alcance uma média maior ou igual a 5,0 (cinco) pontos. A média tem em consideração a nota obtida na avaliação principal (50%) e a nota da prova final (50%).

Faltas acima de 25% do número de aulas: reprovado por falta.

CHEFE DO DEPARTAMENTO

NOME	ASSINATURA
Laura Silva de Assis	

PROFESSOR RESPONSÁVEL PELA DISCIPLINA

NOME	ASSINATURA
Luís Domingues Tomé Jardim Tarrataca	

APROVADO PELO CONSELHO DEPARTAMENTAL EM:

___/___/___

PROGRAMA

- Problemas de Ordenação;
- Critérios de correção;
- Algoritmos de divisão e conquista;
- Crescimento de Funções (limites assintóticos, Notações: Big-O, Little-o , Big-Omega, Little-omega e Teta);
- Recorrências;

- Método de substituição;
 - Método de árvore de recursão;
 - Teorema Mestre.
- Algoritmos de ordenação linear;
 - Tempo linear;
 - Tempo polinomial;
 - Tempo quasi-linear;
- Funções de Hashing;
- Programação Dinâmica:
 - Problema dos números de Fibonacci;
 - Problema dos caminhos mais curtos;
 - Problema de justificação de Textos (sufixos)
 - Problema do Blackjack (sufixos)
 - Problema e Parentização (substrings)
 - Problema da distância de edição (sufixos);
 - Problema da Mochila (sufixos) – tempo pseudo-polinomial;
 - Problema das notas musicais;
 - Problema do Tetris;
- Programação Gulosa:
 - Propriedade Gulosa;
 - Minimum Spanning Trees;
 - Algoritmo de Prim;
 - Single-source shortest paths;
 - Algoritmo de Dijkstra;
- NP-Completeness:
 - Classe P, NP, NP-Hard, NP-Complete;
 - Problema: Super Mario Bros;
 - Problema: 3D Matching;
 - Problema: Subset sum problem;
 - Problema: Partition;
 - Problema: Rectangle packing
- Problema: Jigsaw Puzzles;