

Apresentação da Disciplina

Algoritmos e Estruturas de Dados II

Contextualização

- **Algoritmos e Estruturas de Dados I**: Representação e armazenamento de dados. Manipulação e movimentação de dados em memória principal e secundária. Abstração de dados. Estruturas e abstração de controle. Modularização, Encapsulamento e Herança. Recursividade. Documentação e testes. Implementação em linguagem de programação.
- **Introdução à Computação**: Pesquisa, Ensino, Extensão, Inovação e Mercado. História da computação. Fundamentos de Computabilidade. Linguagens de programação. Principais áreas da Computação. Computação Verde. Práticas de extensão.
- **Cálculo I**: Funções: polinomiais, racionais, algébricas, exponenciais, logarítmicas, trigonométricas. Limites. Continuidade. Derivada: definição e interpretações. Regras de derivação. Derivação implícita. Aplicações da derivada: taxas relacionadas, regra de L'Hospital, estudo do comportamento de funções, esboço de gráficos e otimização. Aplicações práticas.

Ementa

- Somatórios
- Fundamentos de análise de algoritmos
- Ordenação e pesquisa em memória principal
- Tipos abstratos de dados lineares e flexíveis
- Árvores
- Balanceamento de árvores
- Tabelas e dicionários

Objetivos (1/2)

- Fazer com que o aluno desenvolva habilidade de construir programas eficientes por meio da estruturação de dados e da aplicação de algoritmos de ordenação e pesquisa em memória principal
- Propiciar um ambiente no qual o aluno avance no desenvolvimento das habilidades de construção, teste e documentação de programas
- Dar condições para que o aluno desenvolva competências para comparar criticamente diversas soluções para problemas computacionais

Objetivos (2/2)

- Levar o aluno a compreender os aspectos mais importantes da manipulação de dados em memória dinâmica
- Possibilitar a integração das disciplinas do núcleo de matemática e programação de computadores, por meio do desenvolvimento de problemas práticos

Métodos Didáticos (1/2)

- Aulas expositivas com apresentação de conteúdo e discussão de problemas e aplicações
- Desenvolvimento de algoritmos de forma dinâmica durante as aulas
- Revisões de exemplos e atividades práticas que possam estimular o desenvolvimento de uma análise crítica das diversas técnicas estudadas
- Exercícios extraclasse, provas e trabalhos práticos individuais e em grupos

Métodos Didáticos (2/2)

- Aprendizagem por meio de solução de problemas
- Estudos-de-casos que realcem a importância da disciplina e sua aplicação em problemas reais
- Trabalhos práticos e exercícios extraclasse para aprendizado aprofundado dos conceitos e técnicas estudadas

Unidades de Ensino

- Unidade 0: Nivelamento (EAD)
- Unidade I: Fundamentos de análise de algoritmos
- Unidade II: Estruturas de dados básicas lineares
- Unidade III: Ordenação em memória principal
- Unidade IV: Estruturas de dados básicas flexíveis
- Unidade V: Árvores binárias
- Unidade VI: Balanceamento de árvores
- Unidade VII: Tabelas e dicionários
- Unidade VIII: Árvores TRIE

Unidades de Ensino

- **Unidade 0: Nivelamento (EAD)**

- Unidade I: Fundamen

- Unidade II: Estruturas

- Unidade III: Ordenaçã

- Unidade IV: Estrutura

- Unidade V: Árvores b

- Unidade VI: Balancea

- Unidade VII: Tabelas e

- Unidade VIII: Árvores TRIE

- Revisão de AEDs I
- **Introdução ao Linux (vídeo)**
- Introdução às linguagens C, C++ e Java
- Arquivos em C, C++ e Java
- Introdução à Orientação por Objetos
- Recursividade
- Tratamento de exceção
- Ponteiros e referência
- Argumento do Método Main
- Encoding
- Redirecionamento de entrada e saída
- **Processo AEDs II de Exercícios (vídeo)**

Unidades de Ensino

- Unidade 0: Nivelamento (EAD)
 - **Unidade I: Fundamentos de análise de algoritmos**
 - Unidade II: Estruturas
 - Unidade III: Ordenação
 - Unidade IV: Estrutura
 - Unidade V: Árvores b
 - Unidade VI: Balancea
 - Unidade VII: Tabelas e
 - Unidade VIII: Árvores TRIE
- Plano de Ensino
 - **Noções de complexidade (híbrido)**
 - **Pesquisa sequencial e binária (vídeo)**
 - **Algoritmo de ordenação por seleção (híbrido)**
 - **Somatórios (híbrido)**
 - **Contagem de operações (híbrido)**
 - **Aspectos da análise de algoritmos (híbrido)**
 - **Função de complexidade (híbrido)**
 - **Notações O , Ω e Θ (híbrido)**

Unidades de Ensino

- Unidade 0: Nivelamento (EAD)
- Unidade I: Fundamentos de análise de algoritmos
- **Unidade II: Estruturas de dados básicas lineares**
- Unidade III: Ordenação
 - **Lista (híbrido)**
 - Pilha
 - Fila circular
- Unidade IV: Estrutura
- Unidade V: Árvores binárias
- Unidade VI: Balanceamento de árvores
- Unidade VII: Tabelas e dicionários
- Unidade VIII: Árvores TRIE

Unidades de Ensino

- Unidade 0: Nivelamento (EAD)
- Unidade I: Fundamentos de análise de algoritmos
- Unidade II: Estruturas de dados básicas lineares
- **Unidade III: Ordenação**
 - Método da bolha (vídeo)
 - Método de Inserção (híbrido)
 - Shellsort (vídeo)
 - Quicksort (híbrido)
 - Heapsort (híbrido)
 - Mergesort (híbrido)
 - Countingsort (vídeo)
 - Radixsort (vídeo)
 - Comparação entre os métodos (híbrido)
 - Ordenação Parcial (vídeo)
 - Algoritmos Paralelos de Ordenação (vídeo)
- Unidade IV: Estruturas de dados não lineares
- Unidade V: Árvores binárias
- Unidade VI: Balanceamento de árvores
- Unidade VII: Tabelas de hash
- Unidade VIII: Árvores B e B+

Unidades de Ensino

- Unidade 0: Nivelamento (EAD)
 - Unidade I: Fundamentos de análise de algoritmos
 - Unidade II: Estruturas de dados básicas lineares
 - Unidade III: Ordenação em memória principal
 - **Unidade IV: Estruturas de dados básicas flexíveis**
 - Unidade V: Árvores b
 - Unidade VI: Balancea
 - Unidade VII: Tabelas e
 - Unidade VIII: Árvores
- Pilha em Java
 - Fila em Java
 - **Lista simples e dupla em Java (híbrido)**
 - Matriz
 - Pilha, fila e lista em C
 - Coleta de lixo em Java
 - TADs Nativos em Java

- Unidade 0: Nivelamento
- Unidade I: Fundamentos
- Unidade II: Estruturas
- Unidade III: Ordenação
- Unidade IV: Estruturas

- **Definições e conceitos (híbrido)**
- **Tipo Nó em Java (híbrido)**
- Inserção em Java com retorno de referência
- Pesquisa
- Remoção
- Caminhamento
- Inserção em Java com passagem de pai
- Inserção em C com ponteiro
- Inserção em C++ com passagem por referência
- Estruturas híbridas

- **Unidade V: Árvores binárias**
- Unidade VI: Balanceamento de árvores
- Unidade VII: Tabelas e dicionários
- Unidade VIII: Árvores TRIE

Unidades de Ensino

- Unidade 0: Nivelamento (EAD)
- Unidade I: Fundamentos de análise de algoritmos
- Unidade II: Estruturas de dados básicas lineares
- Unidade III: Ordenação
- Unidade IV: Estruturas de dados não lineares
- Unidade V: Árvores b

- **Tipos de rotação (híbrido)**
- **Árvore AVL (híbrido)**
- Árvore 2-3-4
- Árvore Bicolor

- **Unidade VI: Balanceamento de árvores**
- Unidade VII: Tabelas e dicionários
- Unidade VIII: Árvores TRIE

Unidades de Ensino

- Unidade 0: Nivelamento (EAD)
- Unidade I: Fundamentos de análise de algoritmos
- Unidade II: Estruturas de dados básicas lineares
- Unidade III: Ordenação
 - **Tabela Hash Direta com Reserva (híbrido)**
 - Tabela Hash Direta com Rehash
 - Tabela Hash Indireta com Lista
 - Dicionários, conjuntos e mapas
- Unidade IV: Estruturas de dados não lineares
- Unidade V: Árvores binárias
- Unidade VI: Balanceamento de árvores
- **Unidade VII: Tabelas e dicionários**
- Unidade VIII: Árvores TRIE

Unidades de Ensino

- Unidade 0: Nivelamento (EAD)
- Unidade I: Fundamentos de análise de algoritmos
- Unidade II: Estruturas de dados básicas lineares
- Unidade III: Ordenação
 - Tipo Nó
 - Lista flexível
 - Árvore Balanceada
 - Tabela Hash
 - Árvore TRIE
 - Árvore TRIE PATRICIA
- Unidade IV: Estruturas de dados não lineares
- Unidade V: Árvores b
- Unidade VI: Balanceadas
- Unidade VII: Tabelas e dicionários
- **Unidade VIII: Árvores TRIE**

Métodos de Avaliação

- Prova P1, P2 e P3 – 20 pontos cada, sendo que cada prova é dividida em teórica (10 pts), prática (8 pts) e quizzes (2 pts)
- ADA – 5 pontos
- Trabalhos – 20 pontos
- Fator Quiz (FQ) – quizzes a serem usados no desempenho [Nota entre 1 e 1.3]
- Desempenho – 15 pontos (máximo)

$$\text{DESEMPENHO} = \frac{P_{\text{MAIOR}} + P_{\text{SEGUNDAMAIOR}}}{40} \times \frac{\text{Trabalhos}}{20} \times FQ \times 15$$

- Reavaliação – 20 pontos para substituir a nota da menor prova sendo que o aluno aprovado na reavaliação terá nota igual a 60

Exemplo (1): Fator Quiz & Desempenho

- Prova 1: Teoria (5 pts) + Prática (4 pts) + Quizzes (1 pts) = 10 pts
- Prova 2: Teoria (7.5 pts) + Prática (6 pts) + Quizzes (1.5 pts) = 15 pts
- Prova 3: Teoria (10 pts) + Prática (8 pts) + Quizzes (2 pts) = 20 pts
- Trabalhos: 10 pts
- Fator Quiz: $[(1+1.5+2)/6] = [4.5/6] = 75\%$, logo, $1 + 0.3 \times 75\% = 1.23$
- Desempenho = $[(15+20)/40] \times [10/20] \times 1.23 \times 15 = 0.88 \times 0.5 \times 1.23 \times 15 = 8.07$

Exemplo (2): Fator Quiz & Desempenho

- Prova 1: Teoria (5 pts) + Prática (4 pts) + Quizzes (1 pts) = 10 pts
- Prova 2: Teoria (7.5 pts) + Prática (6 pts) + Quizzes (1.5 pts) = 15 pts
- Prova 3: Teoria (10 pts) + Prática (8 pts) + Quizzes (2 pts) = 20 pts
- Trabalhos: **16 pts**
- Fator Quiz: $[(1+1.5+2)/6] = [4.5/6] = 75\%$, logo, $1 + 0.3 \times 75\% = 1.23$
- Desempenho = $[(15+20)/40] \times [16/20] \times 1.23 \times 15 = 0.88 \times 0.8 \times 1.23 \times 15 = 12.92$

Exemplo (3): Fator Quiz & Desempenho

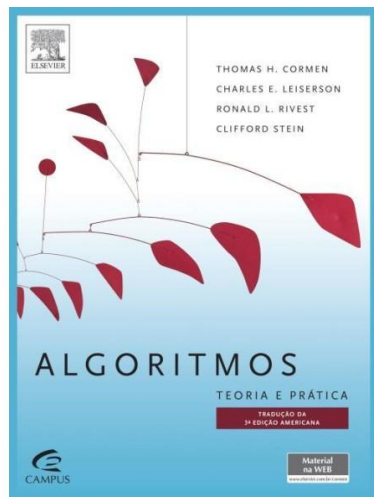
- Prova 1: Teoria (5.5 pts) + Prática (4 pts) + Quizzes (0.5 pts) = 10 pts
- Prova 2: Teoria (8.5 pts) + Prática (6 pts) + Quizzes (0.5 pts) = 15 pts
- Prova 3: Teoria (9.5 pts) + Prática (8 pts) + Quizzes (0.5 pts) = 18 pts
- Trabalhos: 12 pts
- Fator Quiz: $[(0.5+0.5+0.5)/6] = [1.5/6] = 25\%$, logo, $1 + 0.3 \times 25\% = 1.08$
- Desempenho = $[(15+18)/40] \times [12/20] \times 1.08 \times 15 = 0.83 \times 0.6 \times 1.08 \times 15 = 8.07$

Exemplo (4): Fator Quiz & Desempenho

- Prova 1: Teoria (10 pts) + Prática (8 pts) + Quizzes (2 pts)
- Prova 2: Teoria (10 pts) + Prática (8 pts) + Quizzes (2 pts)
- Prova 3: Teoria (10 pts) + Prática (8 pts) + Quizzes (2 pts)
- Trabalhos: **0 pts**
- Fator Quiz: $0.3 \times [(2+2+2)/6] = 1.3$
- Desempenho = $[(P1 + P2)/40] \times [0/20] \times 1.3 \times 15 = 0$

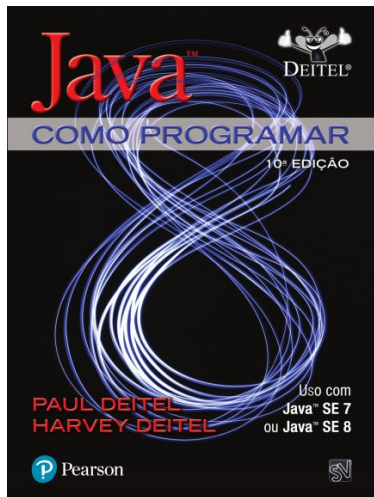
Bibliografia

- CORMEN, T.H., LEISERSON, C.E., RIVEST, R.L, STEIN, C.; Algoritmos: Teoria e Prática; Editora Campus; 3ª Edição; 2012



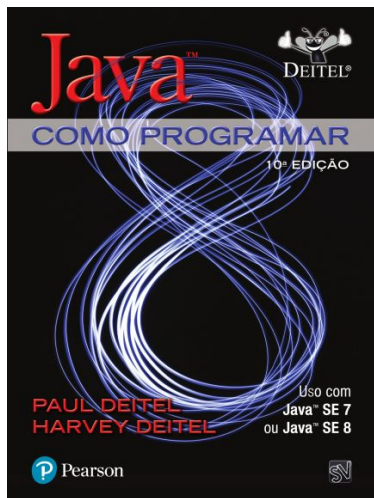
Bibliografia

- DEITEL, H. M.; DEITEL, P. J. Java: Como programar. 10ª edição. Pearson Prentice Hall, 2016



Bibliografia

- DEITEL, H. M.; DEITEL, P. J. Java: Como programar. 10ª edição. Pearson Prentice Hall, 2016



Bibliografia

- ZIVIANI, N. Projeto de algoritmos: Com implementações em Java e C++. Pioneira Thomson Learning, 2006



Bibliografia

- GRAHAN, J., KNUTH, D., PATASHNIK, O.; Matemática Concreta: Fundamentos para a Ciência da Computação; LTC; 2a edição; 1995



Bibliografia

- SCHILDT, H. C Completo e Total. 3ª edição. Pearson, 1997

