



**UNIVERSIDADE FEDERAL DE UBERLÂNDIA
 FACULDADE DE COMPUTAÇÃO
 COLEGIADO DO CURSO DE CIÊNCIA DA COMPUTAÇÃO**

FICHA DE DISCIPLINA

DISCIPLINA: ALGORITMOS E ESTRUTURAS DE DADOS I

CÓDIGO: GBC024		UNIDADE ACADÊMICA: FACULDADE DE COMPUTAÇÃO		
PERÍODO/SÉRIE: - 2º. Período		CH TOTAL TEÓRICA:	CH TOTAL PRÁTICA:	CH TOTAL:
OBRIGATORIA: (X)	OPTATIVA: ()	60	30	90
NÚCLEO DE FORMAÇÃO: Básico				
PRÉ-REQUISITOS: NÃO HÁ		CÓ-REQUISITOS: NÃO HÁ		

OBJETIVOS

Objetivo Geral

Capacitar o aluno a escrever programas eficientes, em termos de tempo e espaço, utilizando boas práticas de programação, para a solução de problemas baseados em listas.

Objetivos Específicos

- Aplicar o conceito de Tipo Abstrato de Dados.
- Discutir o custo computacional de algoritmos.
- Aplicar adequadamente o uso de listas, filas e pilhas.

EMENTA

Alocação dinâmica de memória. Estruturas dinâmicas: pilhas, filas, listas encadeadas. Tipos abstratos de dados.

DESCRIÇÃO DO PROGRAMA

1. Modularidade em programação

- 1.1 Organizar um programa por meio de funções
- 1.2 Passagem de argumentos para funções: valor e endereço
- 1.3 Passagem de argumentos para programa: `argc` e `argv`
- 1.4 Depuração de programas por meio de um *debugger*

2 Tipos Abstratos de Dados

- 2.1 O que é um Tipo Abstrato de Dados (TADs)
- 2.2 Porque criar TADs: Encapsulamento; Reutilização; Manutenção
- 2.3 Como construir um TADs: Encapsulamento; Coesão e Acoplamento; Organização de funções em vários arquivos-fonte; Geração de código objeto (relocável)

3 Listas

3.1 Introdução

- O que são listas
- Operações básicas sobre listas: criação, destruição, inserção, remoção, busca, número de elementos
- TADs para uma lista
- Aplicações para uma lista

3.2 Implementação

- Estática: Organização contígua: ordenada e não ordenada; Algoritmos para operações básicas; Nó cabeça da lista; Custo dos algoritmos para listas ordenadas e não ordenadas
- Dinâmica
 - ✓ Alocação Dinâmica de Memória
 - Endereços de memória de um programa: área de código e de dados
 - Alocação de endereços pelo compilador
 - Alocação de endereços pelo programador: porque é necessário
 - Função de alocação dinâmica de memória: retorno da função
 - Ponteiros: operações e aritmética
 - Relação entre ponteiros e *arrays*
 - Ponteiros e estruturas
 - ✓ Organização encadeada: ordenada e não ordenada
 - ✓ Algoritmos para operações básicas
 - ✓ Nó cabeça da lista
 - ✓ Custo dos algoritmos para listas ordenadas e não ordenadas

4. Lista Circular

- 4.1. O que é uma lista circular: aplicações
- 4.2. Alocação Estática e Dinâmica
- 4.3. Organização contígua: ordenada e não ordenada
- 4.4. Organização encadeada: ordenada e não ordenada
- 4.5. Duplamente encadeada
- 4.6. Algoritmos para operações básicas
- 4.7. Nó cabeça da lista
- 4.8. Custo dos algoritmos (com e sem nó cabeça)
- 4.9. Contígua (ordenada e não ordenada)
- 4.10. Encadeada e Duplamente Encadeada (ordenada e não ordenada)

5. Fila

- 5.1. O que é uma fila: aplicações

F. L. O.

- 5.2. Alocação Estática e Dinâmica
- 5.3. Organização contígua: ordenada e não ordenada
- 5.4. Organização encadeada: ordenada e não ordenada
 - ✓ Duplamente encadeada
 - ✓ Algoritmos para operações básicas
- 5.5. Nó cabeça da lista
- 5.6. Custo dos algoritmos (com e sem nó cabeça)
- 5.7. Contígua (ordenada e não ordenada)
- 5.8. Encadeada e Duplamente Encadeada (ordenada e não ordenada)
- 6. Pilha
 - 6.1. O que é uma pilha: aplicações
 - 6.2. Pilha e Recursividade: Como eliminar a recursão de um algoritmo: uso explícito de uma pilha
 - 6.3. Alocação Estática e Dinâmica
 - 6.4. Organização contígua: ordenada e não ordenada
 - 6.5. Organização encadeada: ordenada e não ordenada
 - ✓ Duplamente encadeada
 - ✓ Algoritmos para operações básicas
 - 6.6. Nó cabeça da lista
 - 6.7. Custo dos algoritmos (com e sem nó cabeça)
 - 6.8. Contígua (ordenada e não ordenada)
 - 6.9. Encadeada e Duplamente Encadeada (ordenada e não ordenada)

BIBLIOGRAFIA

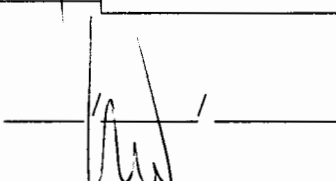
Básica

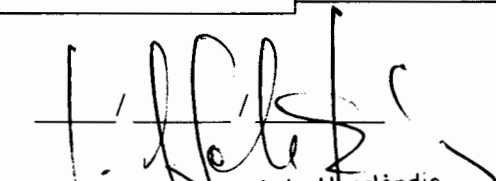
- AHO, A.V & ULLMAN, J.D. & HOPCROFT, J.E. Data Structures and Algorithms. 3a edição, Editora Addison Wesley, 1983
- SEDGEWICK, R. Algorithms in C, Parts 1-4: Data Structures, Sorting, Searching. 3a edição, Editora Addison-Wesley, 1988
- CORMEN, THOMAS H. et. al. Algoritmos: Teoria e Prática. Editora Campus, 2002.

Complementar

- ZIVIANI, N. Projeto de Algoritmos. 2a edição, Editora Thomson, 2004
- TENENBAUM, A.M. & LANGSAM, Y. & AUGENSTEIN, M.J. Estrutura de Dados Usando C. Editora Pearson (Makron Books), 2004

APROVAÇÃO


Universidade Federal de Uberlândia
Carimbo e assinatura do Coordenador do curso
Prof. Ilmério Reis da Silva
Coordenador do Curso de Ciência da Computação
Portaria R nº 713/08


Universidade Federal de Uberlândia
Carimbo e assinatura do Diretor
Prof. Jamil Saraiva
Diretor do Curso de Ciência da Computação
Portaria R nº 672/07