

PLANO DA DISCIPLINA - PROGRAMAÇÃO I: ALGORITMOS E ESTRUTURA DE DADOS

1. IDENTIFICAÇÃO DA DISCIPLINA

- **Nome:** Programação I – Algoritmos e Estrutura de Dados
- **Curso:** Informática e Sistemas de Informação
- **Regime:** Semestral
- **Ano/Semestre:** 3º ano, 1º semestre
- **Carga Horária Total:** 80 horas
- **Horas Semanais:** 5 horas (2 teóricas, 2 teórico-práticas, 1 prática)
- **Precedência Obrigatória:** Introdução à Lógica de Programação

2. OBJETIVOS GERAIS

- Desenvolver competências na representação e manipulação de estruturas de dados.
- Capacitar os alunos na implementação e otimização de algoritmos fundamentais.
- Introduzir os conceitos de árvores e grafos e suas aplicações.
- Familiarizar os estudantes com métodos de ordenação e busca eficientes.

3. COMPETÊNCIAS A DESENVOLVER

- Capacidade de projetar e implementar algoritmos eficientes.
- Conhecimento sobre estruturas de dados fundamentais e suas operações.
- Aplicação de métodos de ordenação e busca.
- Uso de programação recursiva para a resolução de problemas complexos.
- Desenvolvimento de soluções computacionais otimizadas para problemas reais.

4. CONTEÚDO PROGRAMÁTICO

UNIDADE 1: Introdução a Algoritmos e Estruturas de Dados (15h)

- Tipos de dados abstratos.
- Estruturas básicas: Listas, Pilhas e Filas.
- Operações fundamentais sobre listas encadeadas.

UNIDADE 2: Estruturas de Dados Avançadas (20h)

- Árvores binárias e suas representações.
- Árvores balanceadas e árvores binárias de busca.
- Algoritmos de inserção, remoção e busca em árvores.

UNIDADE 3: Algoritmos de Ordenação e Busca (20h)

- Algoritmos de ordenação: Inserção, Intercâmbio, Seleção e Mistura.
- Busca sequencial e busca binária.
- Organização Hash e aplicações.

UNIDADE 4: Grafos e Aplicações (15h)

- Definição e representações de grafos.
- Algoritmos clássicos: Busca em Profundidade e Largura.
- Aplicações de grafos em problemas computacionais.

UNIDADE 5: Laboratórios de Programação (10h)

- Implementação de estruturas de dados em C e Python.
- Desenvolvimento de aplicações com árvores e grafos.
- Simulação e testes de desempenho de algoritmos.

5. METODOLOGIA DE ENSINO

- Aulas expositivas e interativas.
- Exercícios práticos individuais e em grupo.
- Implementação de algoritmos em linguagens de programação.
- Aplicação de estudos de caso e resolução de problemas práticos.

6. CRITÉRIOS DE AVALIAÇÃO

- **Trabalhos práticos e projetos computacionais:** 30%
- **Testes intermediários:** 30%
- **Exame final:** 40%
- **Frequência mínima:** 75% nas aulas práticas.

7. BIBLIOGRAFIA

Bibliografia Básica

- Cormen, T., Leiserson, C., Rivest, R., Stein, C. – *Introduction to Algorithms*, McGraw-Hill e MIT Press, 2001.
- Horowitz, E., Sahni, S. – *Fundamentals of Data Structures*.
- Kernighan, B., Ritchie, D. – *The C Programming Language*, Prentice Hall, 1988.

Bibliografia Complementar

- Knuth, D. – *The Art of Computer Programming: Fundamental Algorithms*, Addison Wesley, 1968.
 - Sedgewick, R. – *Algorithms in C*, Addison-Wesley, 1997.
 - Cairo, Guardati – *Estruturas de Dados*, McGraw-Hill, 1994.
-

