



Plano de Ensino

Dados Gerais

Curso:

217 - Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas (Campus Colinas do Tocantins)

Disciplina:

TCN.0710 - Algoritmos - Graduação [67 h/80 Aulas]

Ano/Período Letivo:

2025/1

Turma:

20251.1.217.1N

Professor(es):

Luis Lima

Data da Aprovação:

29/04/2025

Data da Homologação:

07/05/2025

Ementa

Introdução a algoritmos. Fluxogramas. Teste de mesa. Aspectos do estudo de Linguagens: Sintaxe, Semântica, Paradigmas, Linguagens fonte e objeto, tradução, compilação e interpretação. Introdução à programação de computadores utilizando uma linguagem de programação do paradigma de programação estruturado: variáveis e constantes, aplicação das estruturas de controle e repetição, variáveis indexadas n-dimensionais.

Justificativa

A compreensão de algoritmos e linguagens de programação é fundamental para a formação de profissionais capazes de resolver problemas de maneira lógica, estruturada e eficiente. Esta disciplina propicia o desenvolvimento do raciocínio algorítmico por meio da elaboração de fluxogramas e testes de mesa, fundamentais para a validação e a depuração de soluções computacionais. O estudo dos aspectos teóricos das linguagens de programação — como sintaxe, semântica, paradigmas, tradução, compilação e interpretação — amplia a capacidade crítica e analítica dos estudantes quanto às diferentes abordagens de desenvolvimento de software. Além disso, a introdução prática à programação estruturada, utilizando variáveis, constantes, estruturas de controle, repetição e variáveis indexadas, oferece a base necessária para a construção de programas eficientes e bem organizados.

Objetivo Geral

Desenvolver a capacidade de compreender, modelar e resolver problemas computacionais utilizando conceitos de algoritmos, fluxogramas, teste de mesa e programação estruturada.

Objetivos Específicos

- Compreender o conceito de algoritmo e sua importância na solução de problemas;
- Elaborar fluxogramas e realizar testes de mesa para validar algoritmos;
- Entender os aspectos fundamentais do estudo de linguagens de programação: sintaxe, semântica, paradigmas, linguagens fonte e objeto, tradução, compilação e interpretação;
- Aplicar os conceitos de programação estruturada na construção de programas utilizando variáveis,



constantes, estruturas de controle, estruturas de repetição e variáveis indexadas (vetores e matrizes);

- Desenvolver o raciocínio lógico e a capacidade de abstração na construção de soluções computacionais eficientes;
- Preparar o aluno para o uso consciente e crítico das linguagens de programação como ferramenta de criação e inovação tecnológica.

Conteúdo Programático

1. Introdução a Algoritmos
 - 1.1 Definição e importância dos algoritmos.
 - 1.2 Elementos de um algoritmo: entrada, processamento e saída.
 - 1.3 Características de um bom algoritmo.
- 2 Fluxogramas
 - 2.1 Símbolos e regras de construção.
 - 2.2 Representação gráfica de algoritmos.
 - 2.3 Exercícios práticos de elaboração de fluxogramas.
- 3 Teste de Mesa
 - 3.1 Conceito e importância do teste de mesa.
 - 3.2 Execução manual de algoritmos para detecção de erros lógicos.
- 4 Aspectos do Estudo de Linguagens de Programação
 - 4.1 Conceitos de sintaxe e semântica.
 - 4.2 Paradigmas de programação: estruturado, orientado a objetos, funcional e lógico (introdução).
 - 4.3 Linguagens fonte e objeto.
 - 4.4 Processos de tradução: compilação e interpretação.
- 5 Introdução à Programação Estruturada
 - 5.1 Conceito de linguagem de programação estruturada.
 - 5.2 Variáveis e constantes: definição e tipos de dados.
 - 5.3 Estruturas de controle de fluxo:
 - 5.4 Condicional simples, composta e aninhada.
 - 5.5 Estruturas de repetição: enquanto, para e faça-enquanto.
 - 5.6 Variáveis indexadas n-dimensionais:
 - 5.7 Vetores (arrays unidimensionais).
 - 5.8 Matrizes (arrays bidimensionais e n-dimensionais).
- 6 Construção de Programas
 - 6.1 Análise e resolução de problemas através de algoritmos e codificação.
 - 6.2 Boas práticas de programação: organização, legibilidade e documentação.

Metodologia

Aulas expositivas dialogadas: apresentação dos conceitos teóricos com incentivo à participação e discussão dos conteúdos.

Aulas práticas em laboratório: desenvolvimento de algoritmos, fluxogramas, testes de mesa e codificação de programas utilizando uma linguagem de programação estruturada.

Resolução de exercícios e estudos de caso: aplicação prática dos conceitos estudados em problemas reais e simulados.

Projetos integradores: desenvolvimento de pequenos projetos que envolvam todas as etapas de construção de um programa, do algoritmo à implementação.

Informações Adicionais



A avaliação será composta por atividades práticas e teóricas, além das atividades contínuas realizadas durante as aulas.

A nota de cada bimestre será calculada da seguinte forma:

$$B1 = (N1 + N2) / 2$$

$$B2 = (N1 + N2) / 2$$

Caso o estudante não atinja a nota mínima para aprovação, serão oferecidas atividades de recuperação paralela ao longo do período letivo. Persistindo a insuficiência, o estudante deverá realizar o exame final para nova oportunidade de aprovação.

Referencia Bibliografica Básica

- BACKES, A. Linguagem C: completa e descomplicada. Rio de Janeiro: Elsevier, 2013.
- DAMAS, L. Linguagem C. 10a ed. Rio de Janeiro: LTC, 2016.

Referencia Bibliografica Complementar

- FORBELLONE, A. L. V.; EBERSPÄCHER, H. F. Lógica de programação: a construção de algoritmos e estruturas de dados. 3a ed. São Paulo: Prentice Hall, 2005.
- GUIMARÃES, Â. de M; LAGES, N. A. de C. Algoritmos e estrutura de dados. Rio de Janeiro: LTC, 2015.
- LOPES, A.; GARCIA G. Introdução à programação: 500 algoritmos resolvidos. Rio de Janeiro: Elsevier, 2002.
- MANZANO, J. A. N. G.; OLIVEIRA, J. F. de. Estudo dirigido de algoritmos. 15a ed. São Paulo: Érica, 2012.
- SCHILDT, H. C – completo e total. 3 a ed. São Paulo: Pearson Makron Books, 1997.