

PLANO DE ENSINO

Professor(es): Leonardo Deliyannis Constantin

Curso: Engenharia Mecânica (Semestral) **Turma:** 3V1

Disciplina: Lógica e Algoritmos	
Vigência: a partir de	Período Letivo: 2026/1
Carga horária: 75h	Código:
Ementa: Descrição de sistemas de computação. Definição de algoritmos e sua representação. Análises de compreensão e descrição de problemas através de uma linguagem de programação e utilização de softwares de alto desempenho. Realização de atividades em laboratório.	

Objetivo Geral

Desenvolver o raciocínio lógico e a capacidade de interpretação de problemas. Saber utilizar os recursos básicos da Linguagem de Programação C para a criação de programas.

Objetivos Específicos

- Construir fluxogramas para representar soluções;
- Compreender e utilizar estruturas de seleção e repetição;
- Saber implementar funções, parâmetros e tipos de retorno;
- Compreender e utilizar parâmetros por valor e referência;
- Manipular informações em vetores, matrizes;

Conteúdos

UNIDADE 1 - Introdução a Algoritmo

- 1.1 Lógica de programação
- 1.2 Tipos de linguagens
- 1.3 Tipos de dados
- 1.4 Variáveis e constantes
- 1.5 Representação gráfica de um algoritmo
- 1.6 Operações matemáticas
- 1.7 Operadores lógicos, aritméticos e relacionais

UNIDADE 2 - Testes Condicionais

- 2.1 Teste simples
- 2.2 Teste composto
- 2.3 Teste aninhado
- 2.4 Comando Case

UNIDADE 3 - Estruturas de Repetição

- 3.1 Condição de parada, contador, acumulador
- 3.2 Comando para
- 3.3 Comando enquanto
- 3.4 Comando enquanto
- 3.5 Interrupção antes de satisfazer a condição de parada
- 3.6 Passagem para a próxima iteração

UNIDADE 4 - Estruturas Homogêneas

- 4.1 Variáveis indexadas
- 4.2 Vetores

- 4.3 Matrizes
- 4.4 Operações com vetores e matrizes

UNIDADE 5 - Divisão de Problemas em Sub-rotinas

- 5.1 Conceitos gerais
- 5.2 Parâmetros e argumentos (valor e referência)
- 5.3 Tipos de uma função
- 5.4 Variáveis externas
- 5.5 Regras de escopo
- 5.6 Protótipo da função
- 5.7 Retorno de uma função
- 5.8 Inicialização de variáveis

Cronograma

Dia	Atividades	N. Aula
13/Fev	Apresentação do plano de ensino. Introdução a algoritmos. Ferramentas e atividades de lógica de programação.	1
20/Fev	Revisão de conceitos de algoritmos. Prática de programação em Python.	2
27/Fev	Prática de programação em Python. Introdução a condicionais.	3
06/Mar	Condicionais (if-else). Prática de programação em Python.	4
13/Mar	Condicionais. Prática de programação em Python.	5
20/Mar	Prática de programação em Python. Laços de repetição. Exercício: FizzBuzz.	6
27/Mar	Correção dos exercícios. Prática de programação em Python. Laços de repetição.	7
10/Abr	Correção de exercícios. Estruturas heterogêneas: vetores.	8
17/Abr	Correção de exercícios. Estruturas heterogêneas: vetores e matrizes. Revisão.	9
24/Abr	Avaliação I (prática, 50% da nota).	10
08/Mai	Divisão de Problemas em Sub-rotinas: protótipo e implementação.	11
15/Mai	Divisão de Problemas em Sub-rotinas: parâmetros e retornos.	12
22/Mai	Divisão de Problemas em Sub-rotinas: variáveis locais e globais.	13
29/Mai	Parâmetros e argumentos.	14
12/Jun	Geometria Computacional (parte I).	15
19/Jun	Geometria Computacional (parte II).	16
20/Jun	Geometria Computacional (parte III).	17
26/Jun	Geometria Computacional (parte IV). Revisão.	18
03/Jul	Avaliação II (prática, 50% da nota).	19
10/Jul	Reavaliação.	20

Relação da disciplina com as demais

Essa disciplina é pré-requisito para Cálculo Numérico e Fundamentos de Automação Industrial.

Desenvolvimento Metodológico

A disciplina será conduzida prioritariamente por meio de metodologias ativas, com ênfase na participação dos estudantes, na resolução de problemas e no desenvolvimento do raciocínio lógico e analítico.

As aulas terão caráter teórico-prático, sendo planejadas para serem autocontidas e contemplarem:

- a apresentação e discussão dos conceitos fundamentais de lógica e algoritmos;
- a análise de problemas típicos da área de engenharia;
- a elaboração de algoritmos no papel, por meio de linguagem natural estruturada, fluxogramas ou pseudocódigo;
- a implementação posterior das soluções em laboratório, utilizando a linguagem Python.

Não serão exigidas atividades obrigatórias extraclasse. Exercícios complementares poderão ser sugeridos

de forma opcional, com o objetivo de aprofundamento dos conteúdos.

O caderno do estudante será considerado o principal instrumento de acompanhamento da disciplina e preparação para as avaliações, especialmente para o registro e desenvolvimento dos algoritmos em nível conceitual.

A dinâmica típica das aulas envolverá:

- a apresentação de um ou mais problemas motivadores;
- a construção coletiva e individual de soluções algorítmicas, com ênfase na correção lógica e na justificativa das decisões tomadas;
- a validação teórica dos algoritmos propostos;
- a implementação prática das soluções em Python, quando pertinente.

Os exercícios desenvolvidos em aula poderão servir como base para as avaliações formais da disciplina.

Metodologia de Avaliação e Reavaliação

A avaliação da disciplina será realizada exclusivamente por meio de provas teórico-práticas, com foco na capacidade do estudante de formular, analisar e implementar algoritmos de forma correta e estruturada.

As avaliações ocorrerão em duas etapas ao longo do semestre:

- Prova Prática I – peso 5,0, abrangendo os conteúdos desenvolvidos na primeira metade do semestre;
- Prova Prática II – peso 5,0, abrangendo os conteúdos desenvolvidos na segunda metade do semestre, podendo incluir a continuidade ou aprofundamento de problemas trabalhados anteriormente.

As avaliações poderão envolver:

- conceitos de algoritmos e programação;
- escrita de algoritmos no papel (linguagem natural estruturada, pseudocódigo ou fluxogramas);
- análise e correção de algoritmos propostos;
- implementação, em Python, de algoritmos previamente formulados e justificados.

A correção das provas priorizará:

- a clareza e correção lógica das soluções;
- a coerência entre o algoritmo proposto e sua implementação;
- a capacidade de justificar as decisões tomadas na construção da solução.

Terão direito à Reavaliação os estudantes que obtiverem média final inferior a 6,0 e apresentarem frequência mínima de 75%.

A Reavaliação será realizada em data definida no cronograma da disciplina, consistirá em uma única abrangendo todo o conteúdo do semestre, terá peso 10,0 e substituirá integralmente todas as avaliações realizadas anteriormente.

Dependência

No projeto do curso não prevê dependência.

Horário Disponível para Atendimento

O docente disponibilizará atendimento presencial nos seguintes horários:

- segundas e sextas-feiras, das 08:00 às 09:20, na Sala 501;
- segundas e terças-feiras, das 14:00 às 17:00, na Sala 501;
- sextas-feiras, das 18:30 às 19:20, na Sala 501.

Outros horários poderão ser agendados mediante solicitação com, no mínimo, 24 horas de antecedência.

Bibliografia Básica

- ARAUJO, Jário. Dominando a linguagem C. Rio de Janeiro: Ciência moderna, 2004. 146 p. ISBN 85-7393-315-1.
- FORBELLONE, André Luiz Villar; EBERSPACHER, Henri Frederico. Lógica de programação: a construção de algoritmos e estruturas de dados. 3. ed. São Paulo: Pearson, 2005. 218 p. ISBN 978-85-7605-024-7.
- MANZANO, José Augusto N. G;
- OLIVEIRA, Jayr Figueiredo de. Estudo dirigido de algoritmos. 12. ed. São Paulo: Erica, 2008.

Bibliografia Complementar

CASTRO, J. Linguagem C na prática. Rio de Janeiro: Ciência moderna, 2008.

LOPES, Anita. Introdução à programação: 500 algoritmos resolvidos. Rio de Janeiro: Elsevier, 2002.

SCHILD, Herbert. C++: guia para iniciantes. Rio de Janeiro: Ciência Moderna, 2002.

VILARIM, Gilvan de Oliveira. Algoritmos: programação para iniciantes. 2. ed. Rio de Janeiro: Ciência Moderna, 2004. 270 p. ISBN 978857393316X.

XAVIER, Gley Fabiano Cardoso. Lógica de programação. 11. ed. São Paulo: Senac São Paulo, 2007.

OBSERVAÇÕES

O plano de ensino poderá sofrer ajustes ao longo do semestre em casos excepcionais, devidamente justificados.

Este plano entra em vigor a partir do início do semestre letivo.

Passo Fundo, 09 de fevereiro de 2026.