

Plano de Ensino

CAMPUS VARGINHA

DISCIPLINA: Linguagens de Programação

CÓDIGO: G08LIPR0.01

Início: **01/2026**

Carga Horária: Total: 30 horas/aula Semanal: 02 aulas/aula Créditos: 02

Natureza: Teórica

Área de Formação - DCN: Profissionalizante

Departamento que oferta a disciplina: Departamento de Computação e Engenharia Civil

Ementa:

Evolução das linguagens de programação; noções de sintaxe e semântica; nomes, vinculações; verificação de tipos; tipos de dados; expressões e instruções de atribuição; estruturas de controle no nível de instrução; subprogramas: ambientes de referências locais, métodos de passagem de parâmetros, etc; tipos abstratos de dados; programação orientada a objetos; tratamento de exceções; linguagens de programação funcionais; linguagens de programação lógicas.

Curso(s)	Período	Eixo	Obrigatória	Optativa
Bacharelado em Sistemas de Informação	7º	Desenvolvimento de Software para Sistemas de Informação	X	

INTERDISCIPLINARIDADES

Prerrequisitos

Programação de Computadores II

Laboratório de Programação de Computadores II

Correquisitos

Laboratório de Linguagens de Programação

Objetivos: *A disciplina deverá possibilitar ao estudante*

- 1 Identificar e nomear os princípios fundamentais de linguagens de programação, como variáveis, estruturas de controle e tipos de dados.
- 2 Compreender a sintaxe básica de uma linguagem de programação.
- 3 Implementar algoritmos básicos usando estruturas de dados comuns.
- 4 Comparar e contrastar diferentes paradigmas de programação, como orientação a objetos e programação funcional.
- 5 Criticar e justificar a seleção de estruturas de dados e algoritmos para resolver um problema complexo.
- 6 Propor novas soluções ou abordagens para desafios de programação específicos.

Unidades de ensino

Carga-horária
Horas/aula

Plano de Ensino

1	1. Aspectos Preliminares 1.1 Razões para estudar conceitos de linguagens de programação 1.2 Domínios de programação 1.3 Critérios de avaliação de linguagens 1.4 Influências no projeto de linguagens 1.5 Categorias de linguagens 1.6 Trade-offs no projeto de linguagens 1.7 Métodos de implementação 1.8 Ambientes de programação	02
2	2. Evolução das Principais Linguagens de Programação 2.1 Plankalkül de Zuse 2.2 Programação de hardware mínima: pseudocódigos 2.3 O IBM 704 e Fortran 2.4 Programação funcional: LISP 2.5 O primeiro passo em direção à sofisticação: ALGOL 60 2.6 Informatizando os registros comerciais: COBOL 2.7 O início do compartilhamento de tempo: BASIC 2.8 Tudo para todos: PL/I 2.9 Duas das primeiras linguagens dinâmicas: APL e SNOBOL 2.10 O início da abstração de dados: SIMULA 67 2.11 Projeto ortogonal: ALGOL 68 2.12 Alguns dos primeiros descendentes dos ALGOLs 2.13 Programação baseada em lógica: Prolog 2.14 O maior esforço de projeto da história: Ada 2.15 Programação orientada a objetos: Smalltalk 2.16 Combinando recursos imperativos e orientados a objetos: C++ 2.17 Uma linguagem orientada a objetos baseada no paradigma imperativo: Java 2.18 Linguagens de scripting 2.19 Uma linguagem baseada em C para o novo milênio: C# 2.20 Linguagens híbridas de marcação/programação	02
3	3. Descrevendo 3.1 Introdução 3.2 O problema geral de descrever sintaxe 3.3 Métodos formais para descrever sintaxe 3.4 Gramáticas de atributos 3.5 Descrevendo o significado de programas: semântica dinâmica	02
4	4. Análise Léxica e Sintática 4.1 Introdução 4.2 Análise léxica 4.3 O problema da análise sintática 4.4 Análise sintática descendente recursiva 4.5 Análise sintática ascendente	02
5	5. Nomes, Vinculações e Escopos	02

Plano de Ensino

	5.1 Introdução 5.2 Nomes 5.3 Variáveis 5.4 O conceito de vinculação 5.5 Escopo 5.6 Escopo e tempo de vida 5.7 Ambientes de referenciamento 5.8 Constantes nomeadas	
6	6. Tipos de Dados 6.1 Introdução 6.2 Tipos de dados primitivos 6.3 Cadeias de caracteres 6.4 Tipos ordinais definidos pelo usuário 6.5 Tipos de matrizes 6.6 Matrizes associativas 6.7 Registros 6.8 Uniões 6.9 Ponteiros e referências 6.10 Verificação de tipos 6.11 Tipagem forte 6.12 Equivalência de tipos 6.13 Teoria e tipos de dados	02
7	7. Expressões e Sentenças de Atribuição 7.1 Introdução 7.2 Expressões aritméticas 7.3 Operadores sobrecarregados 7.4 Conversões de tipos 7.5 Expressões relacionais e booleanas 7.6 Avaliação em curto-circuito 7.7 Sentenças de atribuição 7.8 Atribuição de modo misto	02
8	8. Estruturas de Controle no Nível de Sentença 8.1 Introdução 8.2 Sentenças de seleção 8.3 Sentenças de iteração 8.4 Desvio incondicional 8.5 Comandos protegidos	01
9	9. Subprogramas 9.1 Introdução 9.2 Fundamentos de subprogramas 9.3 Questões de projeto para subprogramas 9.4 Ambientes de referenciamento local 9.5 Métodos de passagem de parâmetros 9.6 Parâmetros que são subprogramas 9.7 Subprogramas sobrecarregados 9.8 Subprogramas genéricos 9.9 Questões de projeto para funções 9.10 Operadores sobrecarregados definidos pelo usuário	02

Plano de Ensino

	9.11 Corrotinas	
10	10. Implementando Subprogramas 10.1 A semântica geral de chamadas e retornos 10.2 Implementando subprogramas “simples” 10.3 Implementando subprogramas com variáveis locais dinâmicas da pilha 10.4 Subprogramas aninhados 10.5 Blocos 10.6 Implementando escopo dinâmico	01
11	11. Tipos de Dados Abstratos e Construções de Encapsulamento 11.1 O conceito de abstração 11.2 Introdução à abstração de dados 11.3 Questões de projeto para tipos de dados abstratos 11.4 Exemplos de linguagem 11.5 Tipos de dados abstratos parametrizados 11.6 Construções de encapsulamento 11.7 Nomeando encapsulamentos	02
12	12. Suporte para a Programação Orientada a Objetos 12.1 Introdução 12.2 Programação orientada a objetos 12.3 Questões de projeto para programação orientada a objetos 12.4 Suporte para programação orientada a objetos em Smalltalk 12.5 Suporte para programação orientada a objetos em C++ 12.6 Suporte para programação orientada a objetos em Java 12.7 Suporte para programação orientada a objetos em C# 12.8 Suporte para programação orientada a objetos em Ada 95 12.9 Suporte para programação orientada a objetos em Ruby 12.10 Implementação de construções orientadas a objetos	02
13	13. Concorrência 13.1 Introdução 13.2 Introdução à concorrência no nível de subprograma 13.3 Semáforos 13.4 Monitores 13.5 Passagem de mensagens 13.6 Suporte de Ada para concorrência 13.7 Linhas de execução em Java 13.8 Linhas de execução em C# 13.9 Concorrência no nível de sentença	02
14	14. Tratamento de Exceções e Tratamento de Eventos 14.1 Introdução ao tratamento de exceções 14.2 Tratamento de exceções em Ada 14.3 Tratamento de exceções em C++ 14.4 Tratamento de exceções em Java 14.5 Introdução ao tratamento de eventos	02

Plano de Ensino

	14.6 Tratamento de eventos com Java	
15	15. Linguagens de Programação Funcional 15.1 Introdução 15.2 Funções matemáticas 15.3 Fundamentos das linguagens de programação funcional 15.4 A primeira linguagem de programação funcional: LISP 15.5 Uma introdução a Scheme 15.6 COMMON LISP 15.7 ML 15.8 Haskell 15.9 Aplicações de linguagens fun 15.10 Uma comparação entre linguagens funcionais e imperativas	02
16	16. Linguagens de Programação Lógica 16.1 Introdução 16.2 Uma breve introdução ao cálculo de predicados 16.3 Cálculo de predicados e prova de teoremas 16.4 Uma visão geral da programação lógica 16.5 As origens do Prolog 16.6 Os elementos básicos do Prolog 16.7 Deficiências do Prolog 16.8 Aplicações de programação lógica	02
Total		30

Bibliografia Básica

1	FINKEL, R. A. Advanced programming language design . Reading: Addison-Wesley, 1996.
2	FISCHER, A. E. & GRODZINSKY, F. S. Anatomy of programming languages . Englewood Cliffs: Prentice-Hall, 1993.
3	SEBESTA, R. Conceitos de linguagens de programação . 9. ed. Porto Alegre: Bookman, 2011.

Bibliografia Complementar

1	CASANOVA, M. A. Programação em logica e a linguagem prolog . São Paulo : E. Blücher, 1987. 461 p.
2	JAZZAYERI, M. & GHEZZI, C. Conceitos de linguagens de programação . 3. ed. Rio de Janeiro: Campus, 1998.
3	VAREJÃO, F. M. Linguagens de programação: Java, C e C++ e outras : conceitos e técnicas . Rio de Janeiro : Elsevier, 2004.
4	VILLAS, M. V. Programação: conceitos, técnicas e linguagens . 9.ed. Rio de Janeiro : Campus, 1987. 195 p.
5	VON WANGENHEIM, Aldo. Conhecendo o Smalltalk: todos os detalhes da melhor linguagem de programação orientada a objetos . Florianópolis : Visual, 2002.