



Apresentação da Disciplina



ERICK GALANI MAZIERO
erick.maziero@ufla.br

Departamento de Ciências da Computação
Universidade Federal de Lavras

A decorative graphic on the left side of the slide, featuring a large cyan hexagon with the number '1' inside. Surrounding this central hexagon are several other hexagons in various shades of blue and cyan, some solid and some outlined. There are also small icons, including a network of nodes and a magnifying glass, interspersed among the hexagons.

1

Plano de Curso

Sempre verifiquem o plano de curso disponível no SIG

- Plano semanal

A decorative graphic on the left side of the slide, featuring a large cyan hexagon with the number '2' inside. Surrounding it are several other hexagons in various shades of blue and cyan, some solid and some outlined. There is also a small icon of a network or molecule structure near the top left.

2

Avaliações

Serão 5 Atividades Avaliativas

Vejam pesos no SIG

A decorative graphic on the left side of the slide, featuring a large cyan hexagon with the number 3 inside. Surrounding it are several smaller hexagons in various shades of blue and cyan, some solid and some outlined. There is also a small icon of a network or molecule structure near the top left and a magnifying glass icon near the bottom left.

3

Recuperação

- Atividade Avaliativa Adicional
- Terão direito apenas os alunos que não atingirem nota 60 na média final

$$\text{Nota Recuperada} = (\text{Nota Final} + \text{PA}) / 2$$



4

Verificação de Presença

Durante Estudo Remoto Emergencial (ERE) não haverá aferição de presença

Em possíveis aulas presenciais, será utilizada lista de chamada



5

Metodologia de Ensino

- Aulas expositivas (videoaulas durante o ERE)
- Utilizem o fórum de dúvidas do Campus Virtual
- Façam as listas de exercício, não apenas as atividades avaliativas...

A decorative graphic on the left side of the slide, featuring a large cyan hexagon with the number 6 inside. Surrounding it are several smaller hexagons in various shades of blue and cyan, some with white outlines. There is also a small icon of a network or molecule structure near the top left.

6

Objetivos

Conhecer bem os principais PARADIGMAS de
LINGUAGENS de PROGRAMAÇÃO

Saber escolher qual a melhor para determinado
problema



7

Quem são vocês

Alguma experiência em programação?

Alguma experiência profissional?

Outros cursos de computação já feitos?

A decorative graphic on the left side of the slide, featuring a cluster of hexagons in various shades of blue and cyan. Some hexagons are solid, while others are outlined. A small network icon with a central node and five radiating lines is positioned near the top left. A magnifying glass icon is located near the bottom left.

8

O professor



Formação

- ◇ Bacharelado, Mestrado e Doutorado na Universidade de São Paulo (USP)
 - Início em 2005
- ◇ Doutorado sanduíche na Universidade de Toronto – Canadá
 - Aprendizado semissupervisionado sem fim em Processamento da Linguagem Natural (PLN)





Linhas de Pesquisa

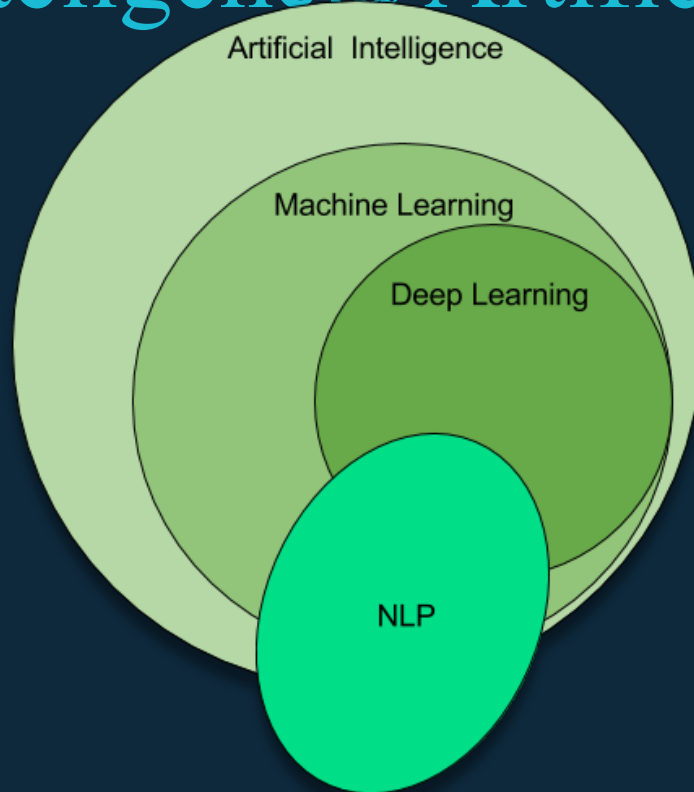
- ◇ Processamento da Linguagem Natural
 - Análise discursiva
 - *Tagging* e *parsing* em geral
- ◇ Inteligência Artificial
 - Aprendizado semissupervisionado
 - Sem fim
 - Incremental
 - Adaptativo



Processamento da Linguagem Natural (PLN)



Inteligência Artificial





8

Materiais da Disciplina

Todo e qualquer material disponibilizado
via Campus Virtual da UFLA

A decorative graphic on the left side of the slide, featuring a large cyan hexagon with the number 9 inside. Surrounding it are several other hexagons in various shades of blue and cyan, some solid and some outlined. There is also a small icon of a network or molecule structure near the top left and a magnifying glass icon near the bottom left.

9

Bibliografia

Consultar plano de Curso
Outras fontes serão indicadas de acordo
com o assunto abordado

A decorative pattern of hexagons in various shades of blue and cyan, some solid and some outlined, arranged in a cluster on the left side of the slide. A small network icon is also visible near the top left.

10

Conceitos Iniciais



O que é um algoritmo

- ◇ Sequência finita de instruções (operações básicas) que levam à solução de um problema.





O que é uma linguagem de programação

- ◇ Conjunto de regras léxicas, sintáticas e semânticas que definem uma linguagem de geração de código-fonte, que, quando compilado ou interpretado, é transformado em um programa de computador



Hierarquia de Chomsky

Linguagem	Gramática	Reconhecedor
[0] Linguagens Recursivamente Enumeráveis	Gramáticas Irrestritas	Máquina de Turing
[1] Linguagens Sensíveis ao Contexto	Gramáticas Sensíveis ao Contexto	Máquina de Turing com Fita Limitada
[2] Linguagens Livres de Contexto	Gramáticas Livres de Contexto	Autômato com Pilha
[3] Linguagens Regulares	Gramáticas Regulares	Autômato Finito





Paradigmas de Linguagem de Programação

- ◇ Modelo, padrão ou estilo de programação suportado por linguagens que agrupam certas características comuns.
- ◇ A classificação de linguagens em paradigmas é influenciada por decisões de projeto - modelagem da solução proposta para um problema real.



A decorative graphic on the left side of the slide, featuring a cluster of hexagons in various shades of blue and cyan. Some hexagons are solid, while others are outlines. A small network icon with a central node and radiating lines is also visible.

11

Resumo dos principais paradigmas



Imperativo

- ◇ Primeiro paradigma a ser definido e conceituado.
- ◇ Segue o conceito de estado e ações que manipulam esse estado.
 - Sequência de instruções que manipulam valores de variáveis.
- ◇ Inclui subrotinas (procedures) para estruturação.
- ◇ Baseado na arquitetura de Von Neumann.
 - Comandos e dados armazenados em memória (a mesma).
 - Instruções, dados e resultados transferidos entre CPU e memória.





Imperativo

- ◇ Subdividido em Estruturado e Não-estruturado
 - Linguagens não estruturadas, em geral, fazem uso de GOTO e JUMP.
 - Linguagens estruturadas não utilizam GOTO e agrupam instruções em blocos.
 - Condicionais.
 - Repetições.
 - Módulos.





Imperativo

◇ Vantagens

- Eficiência.
- Bem estabelecido.
- Modelagem de aplicações reais de modo “natural”.

◇ Desvantagens

- Difícil legibilidade e difícil manutenção.
- Foco no “COMO” e não “O QUÊ”.
- Tratamento de dados e comportamento misturados.





Orientado a objetos

- ◇ Lógica mais próxima do mundo real, onde tudo são objetos.
 - Classes.
 - Objetos.
 - Métodos.
 - Atributos.
- ◇ Esconde o que “não é importante”.
- ◇ Bastante difundido.





Orientado a objetos

Vantagens

- ◇ Maior flexibilidade
 - Herança
 - Sobrecarga
 - Polimorfismo
- ◇ Maior reuso.
- ◇ Alta manutenibilidade.
- ◇ Utilização comercial difundida.





Funcional

- ◇ Baseado em funções matemáticas.
 - Estrutura básica \Rightarrow funções.
 - Qual função deve ser aplicada para transformar uma entrada qualquer na saída desejada.
- ◇ Não são passos sucessivos, e sim funções compostas \Rightarrow aplicações sucessivas de funções.





Funcional

- ◇ Como calcular a distância entre dois pontos?
 - Distância euclidiana

```
euclides(x1, y1, x2, y2)  
= sqrt (((x2 - x1)^2) + ((y2 - y1)^2))
```





Funcional

◇ Vantagens

- Simplifica a resolução de alguns tipos de problemas.
- Concorrência natural.

◇ Desvantagens

- Implementações ineficientes.
- Dificuldade de modelagem de problemas reais.
- Entrada/Saída e formatação primitivos.





Lógico

- ◇ Baseado em lógica formal - Lógica de Predicados de Primeira Ordem.
- ◇ Problema descrito de maneira similar ao raciocínio humano.
- ◇ Declarativo, com características imperativas.
- ◇ Elementos essenciais:
 - Proposições.
 - Regras de inferência.
 - Busca.





Lógico

```
pai (Joao, Maria).  
pai (Joao, Luis).  
pai (Luis, Carlos).  
mae (Joana, Luis).  
mae (Suzana, Carlos).  
progenitor (X,Y) :- mae (X,Y); pai (X,Y).  
pais(X,Y,Z) :- (pai(X,Z), mae(Y,Z)); (pai(Y,Z),  
mae(X,Z)).
```





Lógico

- ◆ Aplicações em IA - sistemas especialistas, sistemas de apoio à decisão, etc.
- ◆ Programas descrevem conjunto de regras que disparam ações quando premissas são satisfeitas.





Lógico

◇ Vantagens

- Conceção da aplicação em alto nível de abstração.
- Mais próximo do raciocínio humano.

◇ Desvantagens

- Implementações ineficientes.
- Dificuldade de modelagem de problemas complexos.
- Entrada/Saída e formatação primitivos.
- Complexidade exponencial.





Scripting

- ◇ São imperativas
- ◇ Mas são de interpretação parcial ou total
- ◇ Vantagens
 - São dinâmicas
 - Servem para resolver problemas rápidos (*scripts*)
- ◇ Desvantagens
 - Não têm muita performance
 - Podem ocorrer muitos erros em tempo de execução





Linguagens de Marcação

- ◇ Não são linguagens de programação
- ◇ Especificam como informações serão dispostas, por exemplo, em documentos Web.
- ◇ Há as linguagens de programação híbridas
 - JSP, por exemplo





Bibliografia

Sebesta, R. W. (2011). *Conceitos de Linguagens de Programação*. 9 ed. Bookman.

