

# Inteligência Artificial

José Mendes 107188

2023/2024



universidade  
de aveiro

---

# 1 Noções de Programação Declarativa

## 1.1 Programação Declarativa

A Programação Declarativa abstrai-se da implementação, focando-se apenas na descrição do que se pretende fazer, enquanto faz uso de dois paradigmas:

- **Programação Funcional**, baseado em funções/cálculo-lambda, a entidade central é a função;
- **Programação em Lógica**, baseado em lógica de primeira ordem, a entidade central é o predicado;

## 1.2 Paradigma Imperativo

O fluxo de operações é especialmente sequenciado de operações com foco na forma como as tarefas são executadas (**instrução**). Podemos **alterar o conteúdo em memória** e ainda (instruções de afetação/atribuição) e ainda **realizar análise de casos** (if-then-else, switch/case, ...), **processamento iterativo** (while, repeat, for, ...) e ter associados **sub-programas** (procedimentos, funções).

**Exemplo:** SQL é uma linguagem declarativa, uma vez que nos seus comandos apenas descrevemos o que queremos obter. Assim, o programador é abstraído da forma como as operações são executadas na prática, ficando essa tarefa a cargo do compilador.

## 1.3 Paradigma Declarativo

	Funcional	Lógico
Fundamentos	Lambda calculus	Lógica de primeira ordem
Conceito central	Função	Predicado
Mecanismos	Aplicação de funções Unificação uni-direccional Estruturas decisórias	Inferência lógica (resolução SLD) Unificação bi-direccional
Programa	Um conjunto de declarações de funções e estruturas de dados	Um conjunto de fórmulas lógicas (factos e regras)

A sua origem data da segunda metade do século XX, mas ainda hoje é amplamente usada e até está em crescimento em áreas como a Inteligência Artificial.

---

## 1.4 Programação Funcional

Possibilidade de definir funções localmente e sem nome. Por exemplo as funções lambda presentes em Lisp e Python.

## 1.5 Programação em Lógica

Um programa é uma teoria sobre um domínio. Por exemplo, temos **socrates é um homem**,  $homem(socrates)$  e, um **homem é mortal**,  $homem(X) : \neg mortal(X)$ . Se perguntarmos se **socrates é mortal**,  $mortal(socrates)$ , a resposta é sim.

## 1.6 Atitude do programador

A programação declarativa, dada a sua elevada expressividade, é pouco compatível com aproximações empíricas (ou seja, tentativa e erro) à programação. Primeiro é preciso pensar bem na estrutura do problema antes de começar a teclar.

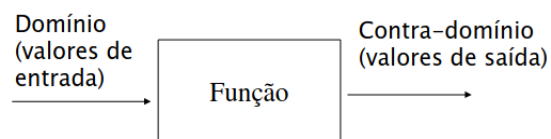
**Passos a seguir:** Perceber o problema  $\rightarrow$  Desenhar o programa  $\rightarrow$  Escrevê-lo  $\rightarrow$  Rever e testar

## 1.7 Características da Programação Funcional

- A entidade central é a função;
- A noção de função é diretamente herdada da matemática (ao contrário das linguagens imperativas, que pode ser muito diferente);
- A estrutura de controlo fundamental é a "aplicação de funções";
- A noção de "tipo da função" captura a noção matemática de domínio (de entrada e saída);
- Os elementos do domínio de entrada e saída podem ser funções;

## 1.8 Função

Tem valores de entrada (domínio) e valores de saída (contradomínio).



## 1.9 Programação em Lógica

Um programa numa linguagem baseada em lógica representa uma teoria sobre um problema. Um programa é uma sequência de frases ou fórmulas representando **factos** (informação sobre objetos do problema/domínio de aplicação) e **regras** (leis gerais sobre esse problema/domínio). Implicitamente as frases estão reunidas numa grande conjunção, e cada frase está universalmente quantificada. Portanto, **programação declarativa**.

---

## 2 Programação ao estilo funcional em Python