

# Linguagens de programação

João G. Santos — RA: 2411612877

## Modelos computacionais

Existem diversos modelos computacionais, sendo o mais comum deles a Máquina de Turing, no qual a computação moderna é massivamente fundada. No entanto existem outros, como Autômatos Celulares, que não possui uso prático, e Cálculo Lambda, que é a base da programação funcional.

mov

## Paradigmas de programação

É possível dividir as linguagens de programação em diferentes paradigmas, cada qual sendo mais apropriado para a resolução de determinados problemas. É incorreto, portanto, afirmar por exemplo que o paradigma *Orientação a objetos* é mais avançado que o *Procedural*, pois enquanto o primeiro é preferido pela indústria de jogos, por exemplo, o segundo é mais popular na construção de kernels e componentes de sistemas operacionais.

Entrementes, vale notar que boa parte das linguagens possuem paradigmas mistos, uma vez que permitem ao desenvolvedor escolher se utilizam um ou outro. Alguns exemplos de exceções são o Java, que é puramente orientado a objetos, e o Haskell, que é puramente funcional.

### Procedural

- **Exemplos:** COBOL, FORTRAN, ALGOL, PL/I, BASIC, C.
- 

#### Orientação a objetos

- **Exemplos:** Java, C++, Smalltalk.

### Declarativo

- **Exemplos:** SQL, Prolog.

### Funcional

- **Exemplos:** Haskell, Lisp, OCaml.

## Programação funcional

Na programação funcional a computação é feita através de funções matemáticas. Uma vez que a programação funcional não admite estados ou dados mutáveis, há muito menos chance de um estado imprevisto provocar um comportamento inesperado (bug), fato pelo qual esse paradigma é bastante usado por empresas do ramo financeiro e de criptomoedas.

O exemplo abaixo cria uma função que computa o dobro de “x”. É bastante visível a similaridade com o equivalente matemático “ $f(x) = x * 2$ ”.

```
f :: Integer -> Integer
f x = x * 2
```

```
main = do
    putStrLn $ fn 2
```