

# Programação funcional

---

Marco A L Barbosa

malbarbo.pro.br

Departamento de Informática

Universidade Estadual de Maringá



Este trabalho está licenciado com uma Licença Creative Commons - Atribuição-Compartilhualgal 4.0 Internacional.

<http://github.com/malbarbo/na-progfun>

O quê é?

O que é programação imperativa?

- Um paradigma de programação que usa sentenças para modificar o estado do programa.

O que é programação funcional?

- Um paradigma de programação que usa aplicação e composição de funções para construir programas (inspirado no cálculo lambda).
- Evita mudança de estado.
- Evita feitos colaterais.

Por quê?

Um paradigma (linguagem) de programação é uma ferramenta.

Conhecer várias ferramentas permite utilizar a mais adequada para cada problema.

Em geral os programas são mais simples, fáceis de entender e modificar.

Diversão! Programar, no paradigma funcional, é muito divertido.

Como?

Escolher uma linguagem.

Ler a fundamentação teórica e as referências da linguagem.

Praticar lendo e escrevendo código.

## Escolher uma linguagem

- Racket
- Bom suporte ao paradigma funcional
- Variante moderna do Lisp
- Ambiente integrado DrRacket
- Documentação extensa
- Fácil instalação

Ler a fundamentação teórica e as referências da linguagem

- [A Tutorial Introduction to the Lambda Calculus](#)
- Livro [How to Design Programs](#)
- [Guia e Referência](#) do Racket
- Livro [Structure and Interpretation of Computer Programs](#)
- Livro [The Scheme Programming Language](#)

Praticar lendo e escrevendo código

- Muitos exemplos
- Muitos exercícios

Primeiros passos

```
$ apt-get install racket
```

```
$ drracket
```

Ficheiro Editar Ver Linguagem Racket Insert Scripts Tabs Ajuda

SemNome 2 ▾ (define ...) ▾  Verificar Sintaxe  Debug  Macro Stepper   Executar  Parar 

```
#lang racket
```

Bemvindo a [DrRacket](#), versão 7.2 [3m].

Linguagem: racket, with debugging; memory limit: 128 MB.

```
> |
```

Determine language from source ▾

3:2

559.93 MB



Ficheiro Editar Ver Linguagem Racket Insert Scripts Tabs Ajuda

step.rkt ▾ (define ...) ▾  Verificar Sintaxe  Debug  Macro Stepper  Executar  Parar 

```
#lang racket
```

```
(define (quadrado x)
  (* x x))
```

## Definições

Bemvindo a [DrRacket](#), versão 7.2 [3m].

Linguagem: racket, with debugging; memory limit: 128 MB.

```
> (quadrado 5)
```

```
25
```

```
>
```

## Interações

Determine language from source ▾

5:2

567.46 MB



Na área de definições

- Digite o código do programa
- Pressione o botão executar/correr (ctrl + r)

Na área de interações

- Teste as definições usando REPL (Read Eval Print Loop)
- Entre com uma expressão (Read)
- A expressão é avaliada (Eval)
- O resultado da expressão é impressa (Print)
- Repetida o processo (Loop)

## Exemplos de interações

```
> 34
```

```
34
```

```
> (+ 3 4)
```

```
7
```

```
> (* 2 6)
```

```
12
```

```
> +
```

```
#<procedure:+>
```

Leitura

## Recomendada

- [Introdução rápida ao Racket](#)
- [Programação funcional](#)

## Extra

- [The Python paradox](#)
- [Revenge of the Nerds](#)
- [Beating the averages](#)