



Código Concorrente em Go

Agosto 2021

Quem sou?

Pedro Ramos

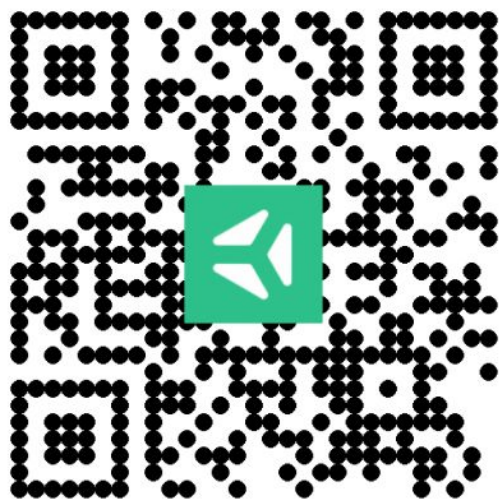
- Bacharel e Mestre em Computação pela UFMG
- Instrutor de Software na **trybe**
- Minhas armas prediletas: Go, Python, Node, C/C++.





Índice

- > Explicação teórica e Motivação (5 min)
- > Demonstração técnica (35 min)
- > Dúvidas (10 min)

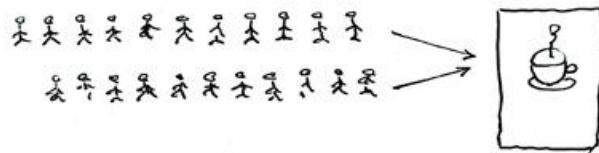


Faça parte
do nosso time.

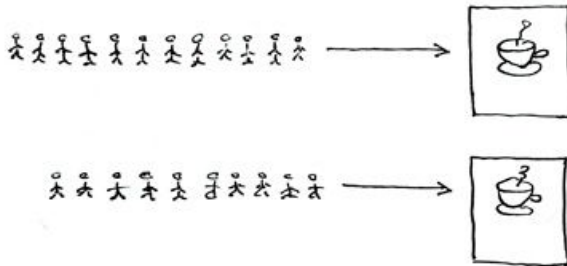


Concorrência VS Paralelismo

Concurrent = Two Queues One Coffee Machine



Parallel = Two Queues Two Coffee Machines



© Joe Armstrong 2013



Porquê escrever código concorrente?

- Servidores web recebem milhares de requisições de clientes de uma só vez
- Apps mobile renderizam animações enquanto simultaneamente executam tarefas e requisições no fundo
- Problemas “batch” tradicionais usam concorrência para esconder a latência de operações de entrada e saída (I/O) (*como NodeJS, que é concorrente mas não é paralelo*)
 - 1. Ler dados
 - 2. Computar
 - 3. Escrever a saída
- Computadores crescem em número de cores, mas não mais em velocidade





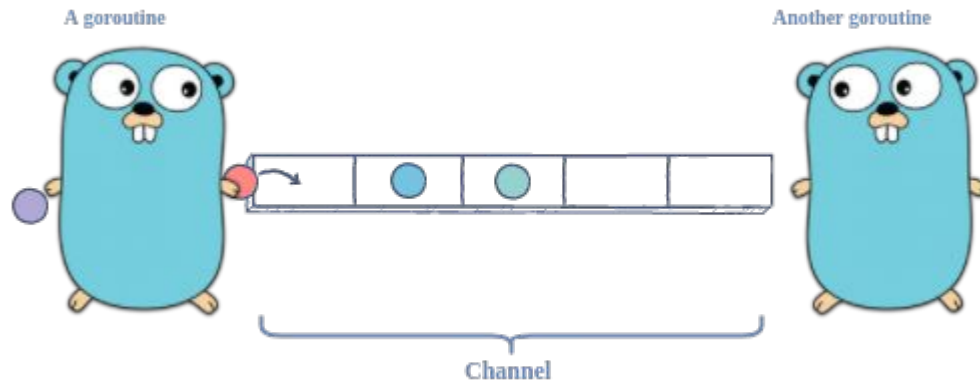
Porquê Go?

- Nov 2009: máximo efeito pelos mínimos caminhos
- Eficiente em Compilação E Execução
- Processamento massivo de dados (único da Google) através da rede
- Gerenciamento de memória
- Fim de heranças
- Concorrência *made easy*
- Propósito genérico: redes, machine learning, mobile, gráficas





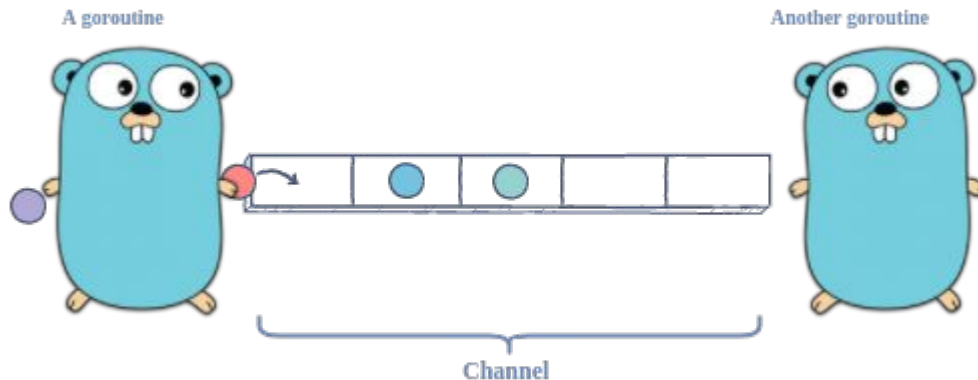
Canais em Go



- CSP (*Communicating Sequential Processes*):
 - Valores passados entre rotinas
 - **Não há estado de memória compartilhado**



Canais em Go



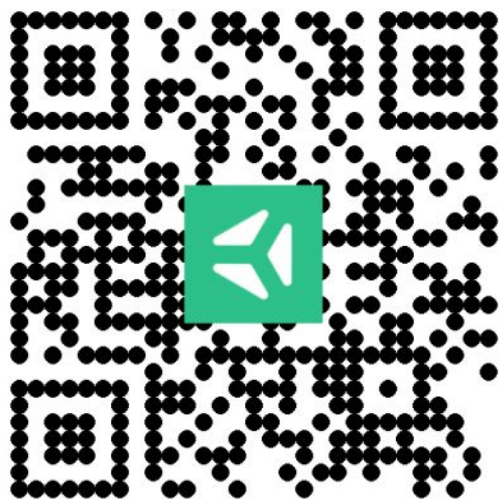
- CSP (*Communicating Sequential Processes*):
 - Valores passados entre rotinas
 - **Não há estado de memória compartilhado**

SINTAXE BLOCANTE!

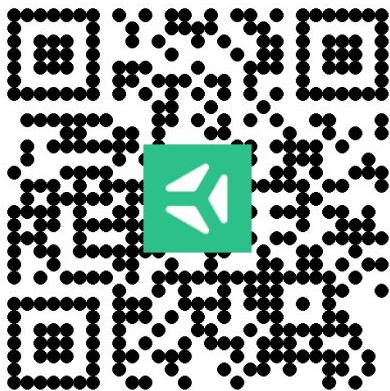


Demonstração

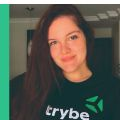
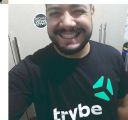
SHOW ME THE CODE!!!



Faça parte
do nosso time.



Gere oportunidades
na vida das pessoas.
Vem pra Trybe!





betrybe.com

