

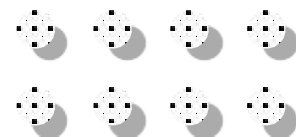
# OBI

## Aula 02

GEMP  
Grupo de Estudos da  
Maratona de Programação

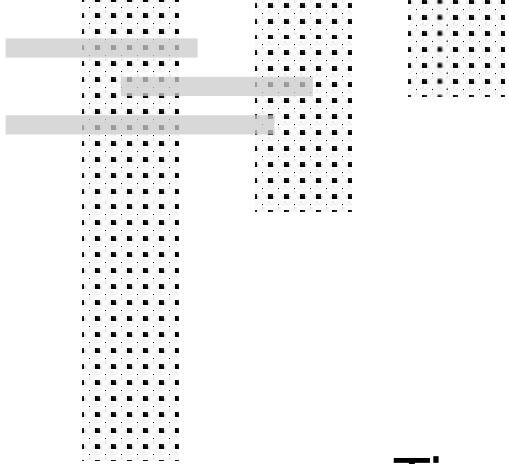
# Tópicos

- O que vimos
- Condicionais
- Loops / Repetições
- Kata
- Kake




# O que vimos...


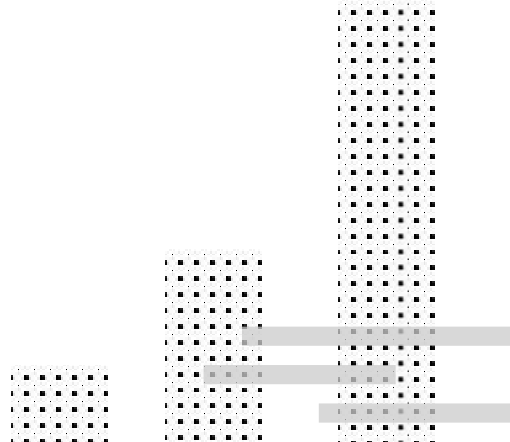
- Utilizaremos C++ nos encontros
- Dúvidas nas linguagens da OBI (Java, JS, Python, C ou C++)
- Dojo:
  - Kata → Exercício de fixação (geralmente já respondido)
  - Randori → Duplas que se alternam no projetor
  - Kake → Duplas em paralelo e que se alternam



# C++



- Tipos:
  - string
  - int
  - float
  - double
  - bool
  - char

- unsigned
  - long
  - short
  - bool
  - auto
  - void\*
- 
- 



C++ hello\_world.cpp

```
#include <iostream>

using namespace std;

void print_hello() {
    cout << "Hello World!" << endl;
}

int main() {

    print_hello();

    return 0;
}
```



CodeImage



C++ hello\_world.cpp

```
#include <iostream>

using namespace std;

void print(string message);

int main() {

    print("Hello World!");

    return 0;
}

void print(string message) {
    cout << message << endl;
}
```



CodelImage



# Condicionais

## **if**

Quando verdadeiro executa uma série de instruções

## **else if**

Um “if” para caso o último “if” ou “else if” fosse falso

## **else**

Quando nenhum “if” ou “else if” foi verdadeiro executar uma série de instruções

## **?: ou ... ? ... : ...**

Operador ternário

## **switch**

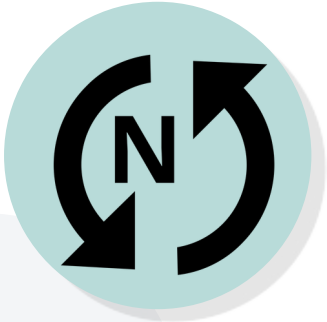
Comparações com base em casos

## **conditions**

`== , != , && , || , > , < , <= , >= , and , or , not , true , false , !`



# Loops



**for**

Rodar por N vezes



**while**

Rodar enquanto houver  
condição



**Recursão**

Rodar até condição de  
parada

# Kata

Fazer um programa capaz de mostrar indefinidamente os algarismos das aproximações de  $\pi$ .

Ex.:

$$\begin{array}{r} 22 \\ \hline 7 \end{array} \quad \begin{array}{r} 355 \\ \hline 113 \end{array} \quad \begin{array}{r} 245850922 \\ \hline 78256779 \end{array}$$

# 01

## Testes

$$\frac{22}{7} = 3.\overline{142857}$$

# 03

## Geral

Servir para qualquer aproximação de pi.

# 02

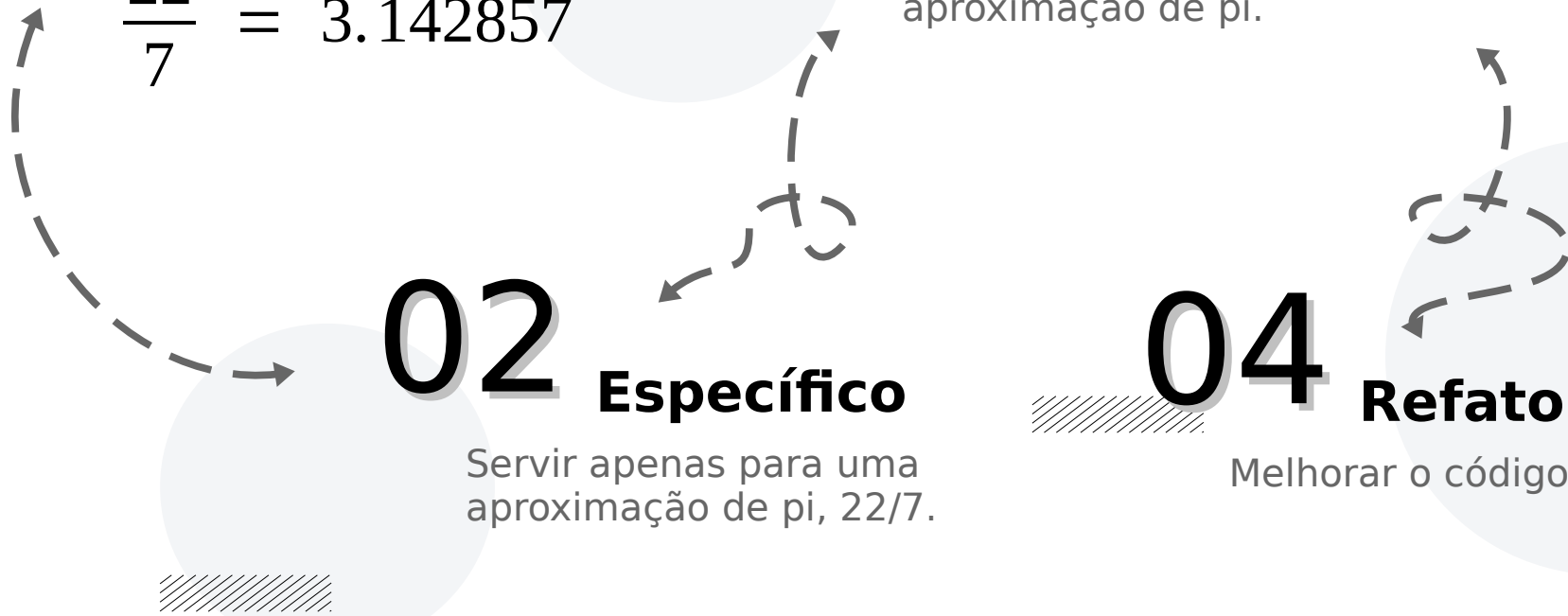
## Específico

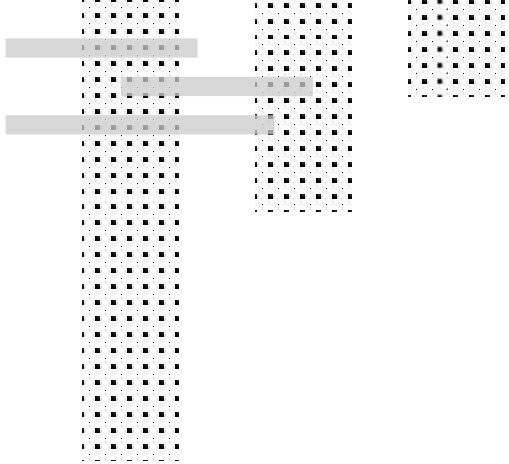
Servir apenas para uma aproximação de pi, 22/7.

# 04

## Refatorar

Melhorar o código





$\pi =$

3.14159 26535 89793 23846 26433 83279 50288 41971 69399 37510...

