


# Estruturas de controle e repetição

## Tomando decisões dentro do código

Prof. Dr. **Walmes Zeviani** · UFPR

2021-05-06

# O que essas estruturas

Existem 2 tipos básicos de estruturas de programação que estão presentes nas linguagens de programação.

## Estruturas de controle

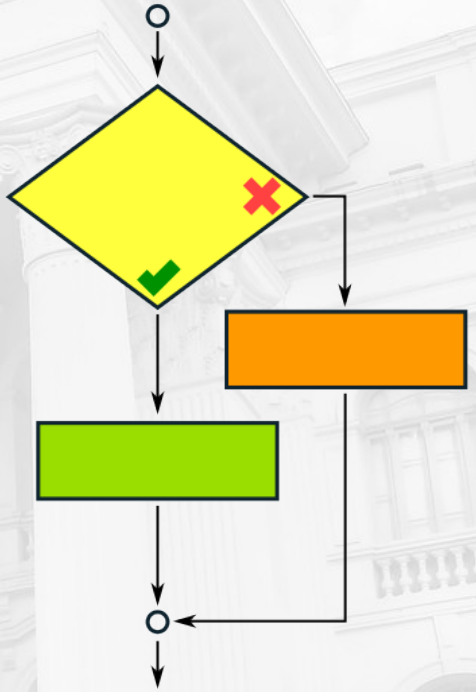
- Elas permitem fazer execução condicional ou seletiva de instruções conforme o resultado de condições que são testadas.
- Os tipos mais comuns são:
  - IF-ELSE.
  - SWITCH ou CASES.

## Estruturas de repetição

- Também chamados de loops ou estruturas de iteração que permitem executar instruções um certo número de vezes até que uma condição seja atingida.
- Os tipos mais comuns são:
  - FOR ou FOREACH.
  - WHILE.
  - REPEAT ou DO-UNTIL.

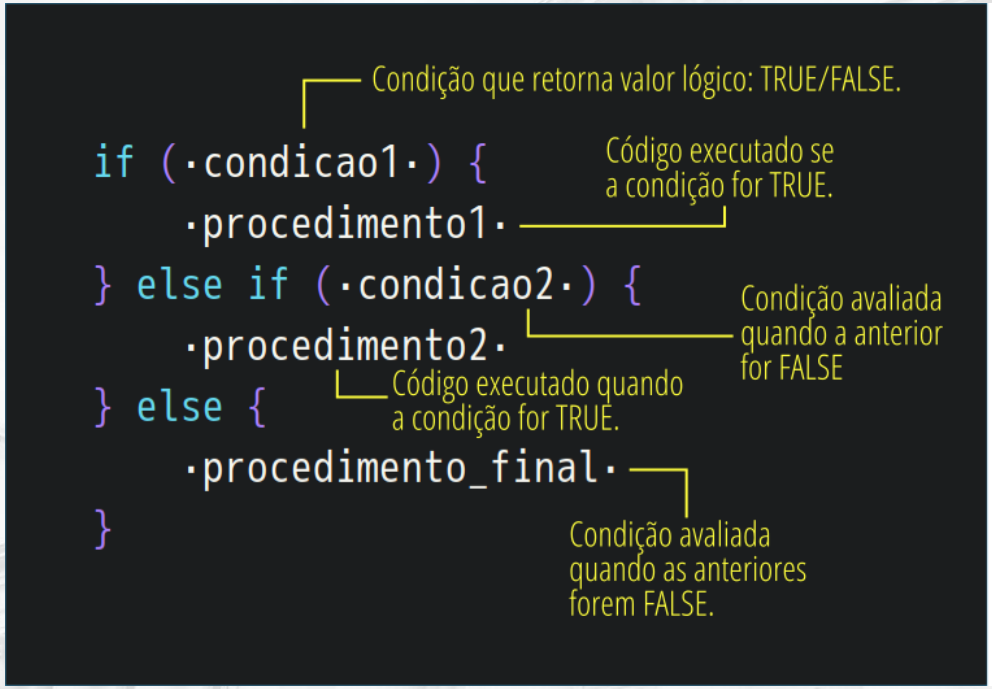
# Estrutura IF-ELSE

- O IF-ELSE permite desviar a execução do código conforme uma condição cujo resultado é lógico, ou seja, TRUE ou FALSE.
- Se o resultado da condição for TRUE, executa-se certas instruções.
- Se o resultado da condição for FALSE, executa-se outras instruções.



Fluxograma de uma estrutura de controle tipo IF-ELSE. Fonte: os autores.

# Exemplo de estrutura IF-ELSE no R

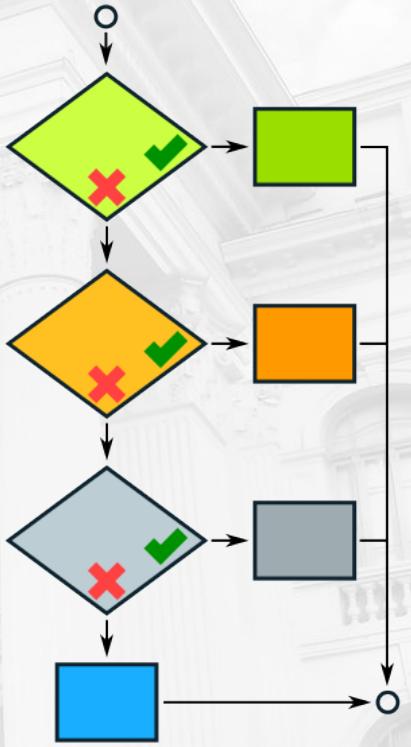



```r
faltas <- 10
nota <- 70
if (nota >= 70 & faltas < 15) {
    result <- "Aprovado"
} else if (nota < 70) {
    result <- "Reprovado por nota"
} else {
    result <- "Reprovado por faltas"
}
result
```

Exemplo de código com a estrutura de controle tipo IF-ELSE. Fonte: os autores.

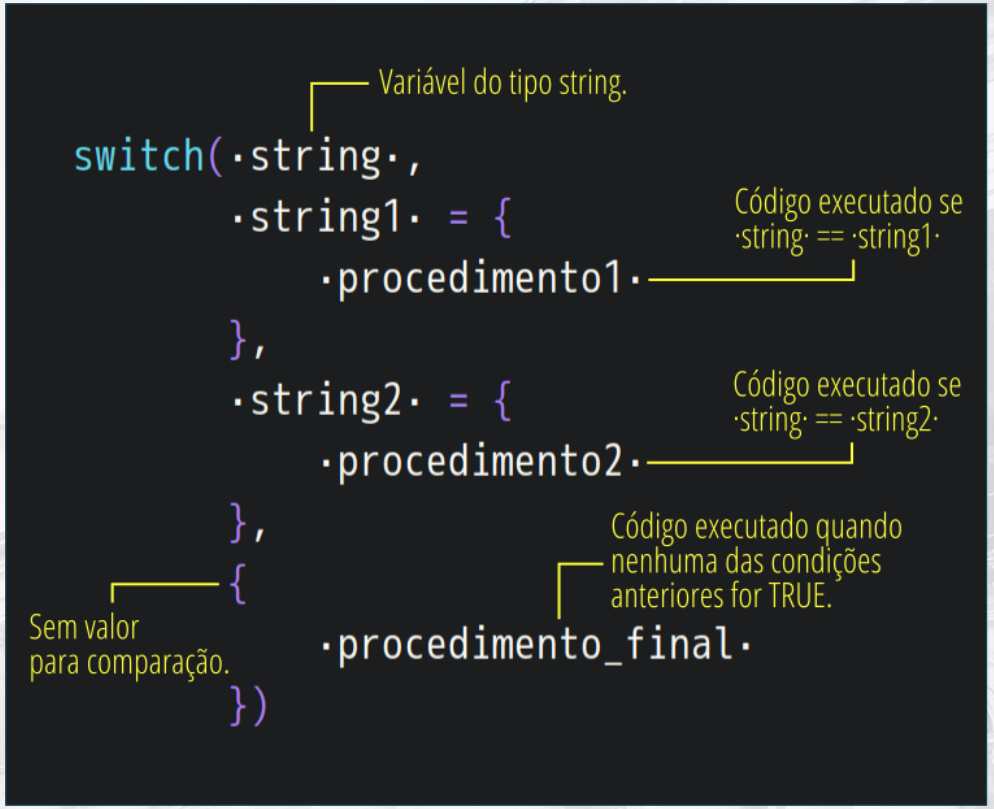
# Estrutura SWITCH

- O SWITCH nada mais é que um empilhamento de estruturas IF-ELSE.
- No entanto, ele confere mais clareza para o código e eficiência[1].
- O caso mais comum é usar SWITCH para testar um valor contra um conjunto de strings.
- Geralmente há uma condição final ou de escape.



Fluxograma de uma estrutura de controle tipo SWITCH. Fonte: os autores.

# Exemplo de estrutura SWITCH no R




```r
animal <- "gato"
som <-
    switch(animal,
           "cachorro" = {
               latir()
           },
           "gato" = {
               miar()
           },
           "vaca" = {
               mugir()
           },
           {
               silencio()
           })
som
```

Exemplo de código com a estrutura de controle tipo SWITCH. Fonte: os autores.

# Exercício · Alíquota do IR

Pege o salário do usuário e determine a alíquota correspondente no imposto de renda. A faixa de salário está na coluna 1 da tabela ao lado e a alíquota está na 2.

1. Use a estrutura IF-ELSE.
2. Use a estrutura SWICTH. Dica, empregue a função `findInterval()`.

**TABELA DE ALÍQUOTA IRPF A DEDUZIR**

Conforme o salário, você confere em qual faixa de contribuição se encaixa

| BASE DO CÁLCULO (R$) | ALÍQUOTA | PARCELA A DEDUZIR DO IRPF (R$) |
|---|---|---|
| Até R$ 1.903,98 | - | - |
| De R$ 1.903,99 a R$ 2.826,65 | 7,5% | R$ 142,80 |
| De R$ 2.826,66 a R$ 3.751,05 | 15% | R$ 354,80 |
| De R$ 3.751,06 a R$ 4.664,68 | 22,5% | R$ 636,13 |
| Acima de R$ 4.664,69 | 27,5% | R$ 869,36 |
| Parcela por dependente | R$ 189,59 | |

Fonte: Medida Provisória 670/2015, convertida na Lei 13.149/2015

Tabela de imposto de renda. Fonte: https://www.leoa.com.br/blog/tabela-do-imposto-de-renda-2020


?
Dúvidas, comentários, sugestões?

# Versões vetoriais

- A estrutura IF-ELSE apresentada não é vetorial, ou seja, no teste da condição são comparados dois valores.
- Quando a condição tiver que ser testada sobre uma série de valores em um vetor, pode-se a função `base::ifelse()` ou `dplyr::if_else()`.
- A versão vetorial para a estrutura SWTICH é o `dplyr::case_when()`.

```
# Notas dos alunos.
notas <- c("João" = 70, "Ana" = 89,
           "Márcia" = 81, "Tiago" = 65,
           "Rodrigo" = 35)

# Usando IF-ELSE vetorial.
ifelse(notas >= 70, "Aprovado",
       ifelse(notas >= 40, "Exame",
              "Reprovado"))
```

```
##        João         Ana
##  "Aprovado"  "Aprovado"
##      Márcia       Tiago
##  "Aprovado"     "Exame"
##     Rodrigo
## "Reprovado"
```

```
# Usando SWITCH vetorial.
dplyr::case_when(notas >= 70 ~ "Aprovado",
                 notas >= 40 ~ "Exame",
                 TRUE ~ "Reprovado")
```

```
## [1] "Aprovado"  "Aprovado"
## [3] "Aprovado"  "Exame"
## [5] "Reprovado"
```