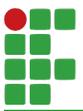


Algoritmos: Fluxo condicional

PRG129001 – Programação I

Prof. Roberto Wanderley da Nóbrega
Instituto Federal de Santa Catarina

2024.1



Estes slides são baseados no material do Prof. Eraldo, disponível [na wiki](#).

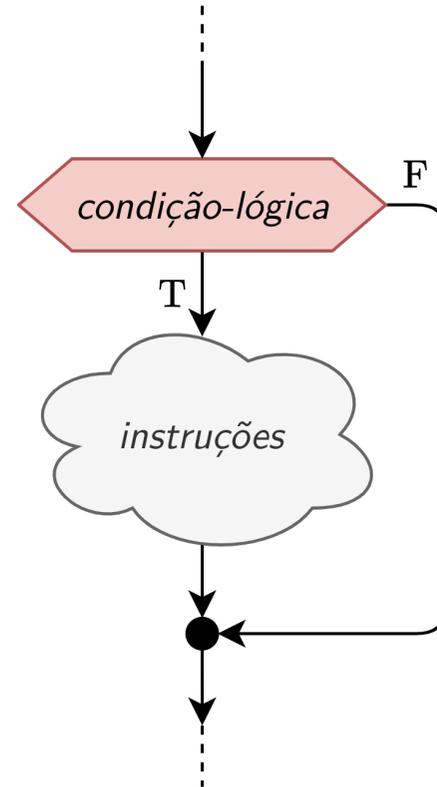


Conceito

Em várias situações, é necessário que o algoritmo tome uma **decisão** sobre qual instrução executar a seguir, com base em uma **expressão lógica**. Para isso, utilizamos blocos ou comandos que permitem o **controle condicional** do fluxo de execução do algoritmo.

Comando Se

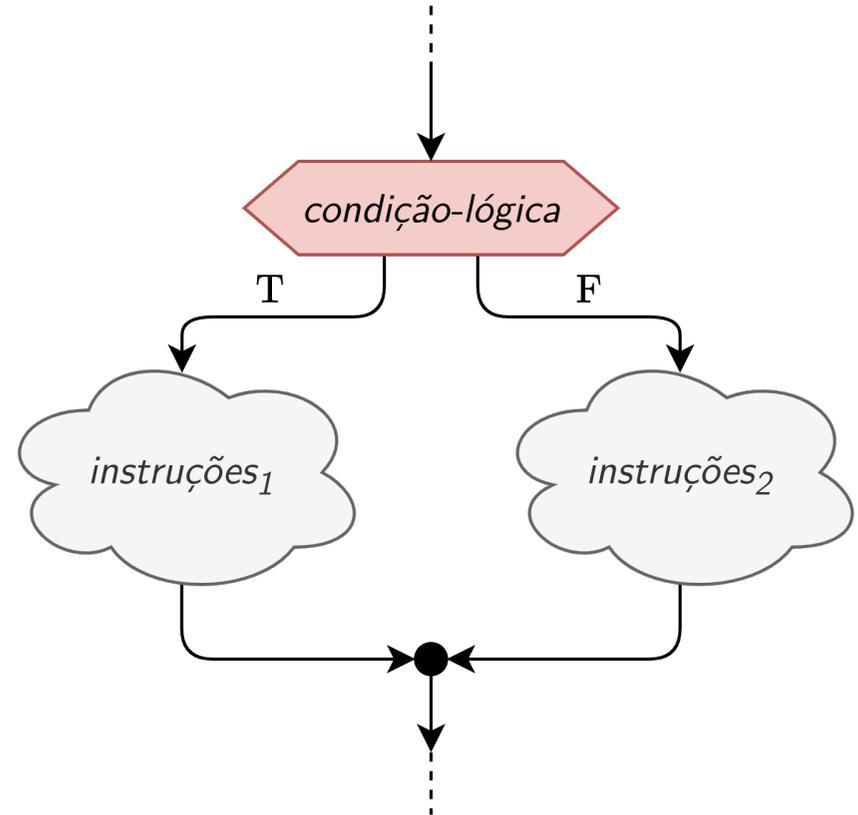
Se *condição-lógica* **então**
instruções
FimSe

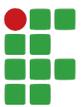




Comando Se-Senão

Se *condição-lógica* **então**
instruções₁
Senão
instruções₂
FimSe



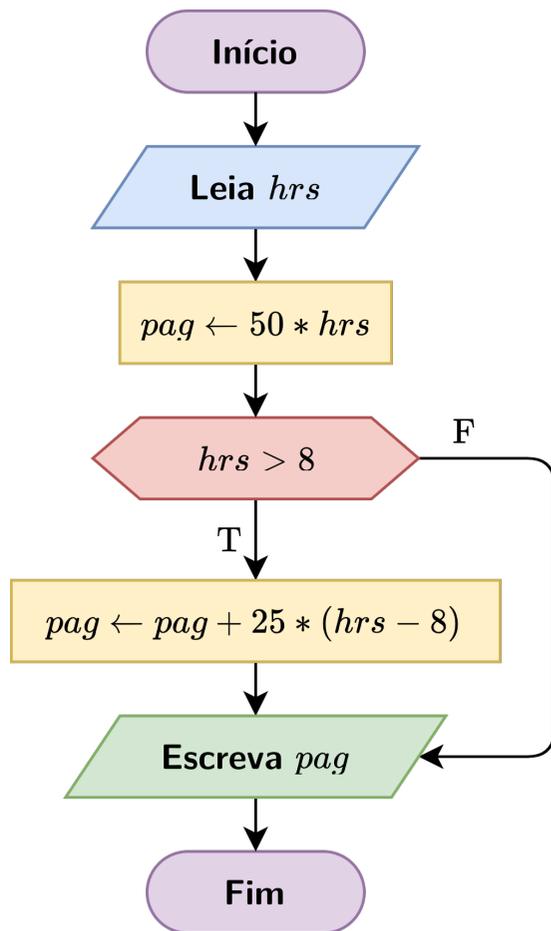


Exemplo: Hora extra

Considere um colaborador de uma empresa que recebe R\$ 50 por hora trabalhada. Porém, se ele trabalhar mais de 8 horas por dia, ele recebe um adicional de R\$ 25 por hora extra trabalhada. Por exemplo:

- Se ele trabalhar 6 horas, ele receberá $50 \times 6 = 300$ reais.
- Se ele trabalhar 10 horas, ele receberá $50 \times 10 + 25 \times 2 = 550$ reais.

Elabore um algoritmo que **LÊ** um número real representando a quantidade de horas trabalhadas em um dia pelo colaborador e **ESCREVE** o valor do seu pagamento ao final do dia.



Variáveis

hrs, pag : real

```
1 Início
2 Leia hrs
3  $pag \leftarrow 50 * hrs$ 
4 Se  $hrs > 8$  então
5      $pag \leftarrow pag + 25 * (hrs - 8)$ 
6 FimSe
7 Escreva pag
8 Fim
```



Exemplo: Aposentadoria

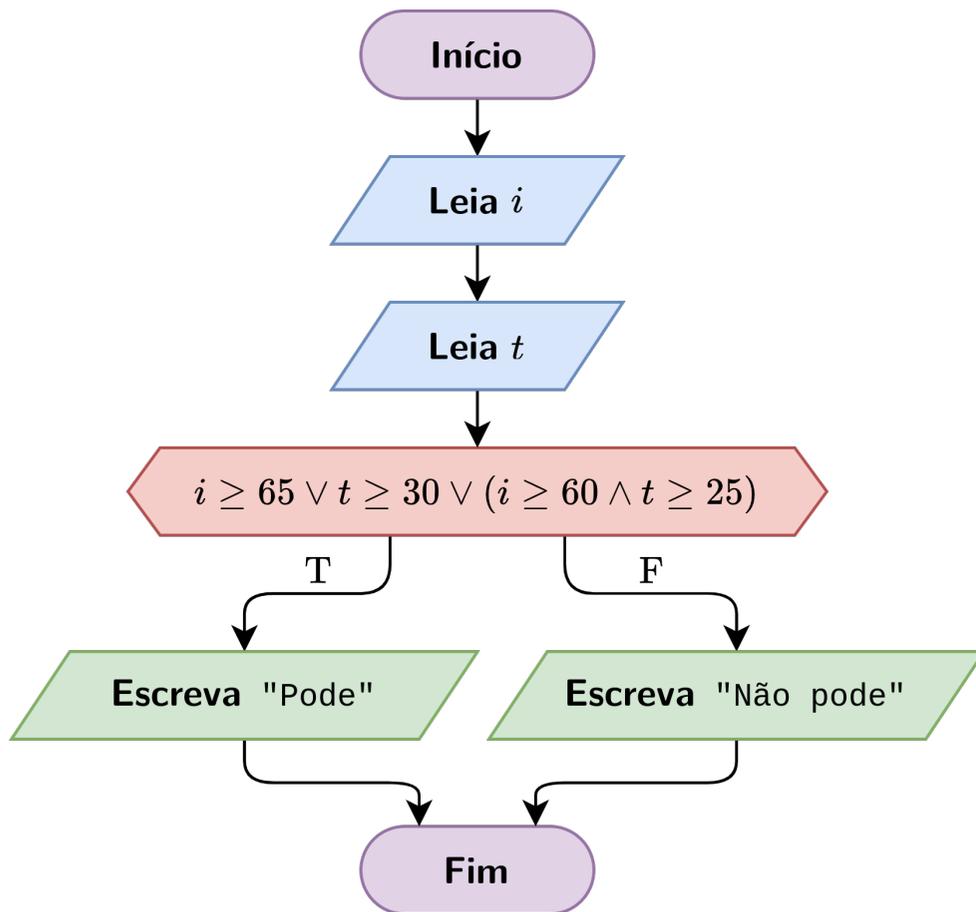
Suponha que, em determinado país, um cidadão pode se aposentar se satisfizer pelo menos uma das seguintes condições:

- Ter pelo menos 65 anos.
- Ter trabalhado pelo menos 30 anos.
- Ter pelo menos 60 anos e trabalhado pelo menos 25 anos.

(a) Escreva uma expressão lógica que, dadas as variáveis inteiras i e t , representando a idade e o tempo de serviço, assume o valor \mathbb{T} , se o cidadão pode se aposentar, ou \mathbb{F} , caso contrário.

$$i \geq 65 \vee t \geq 30 \vee (i \geq 60 \wedge t \geq 25)$$

(b) Elabore um algoritmo que **LÊ** dois números inteiros representando a idade e o tempo de serviço de um cidadão e **ESCREVE** a mensagem **Pode**, se o cidadão pode se aposentar, ou a mensagem **Não pode**, caso contrário.



Variáveis

i : inteiro (*idade*)

t : inteiro (*tempo de serviço*)

Obs.: A abreviação do nome de variáveis não é uma prática recomendada; estamos abreviando aqui apenas por questões de espaço.

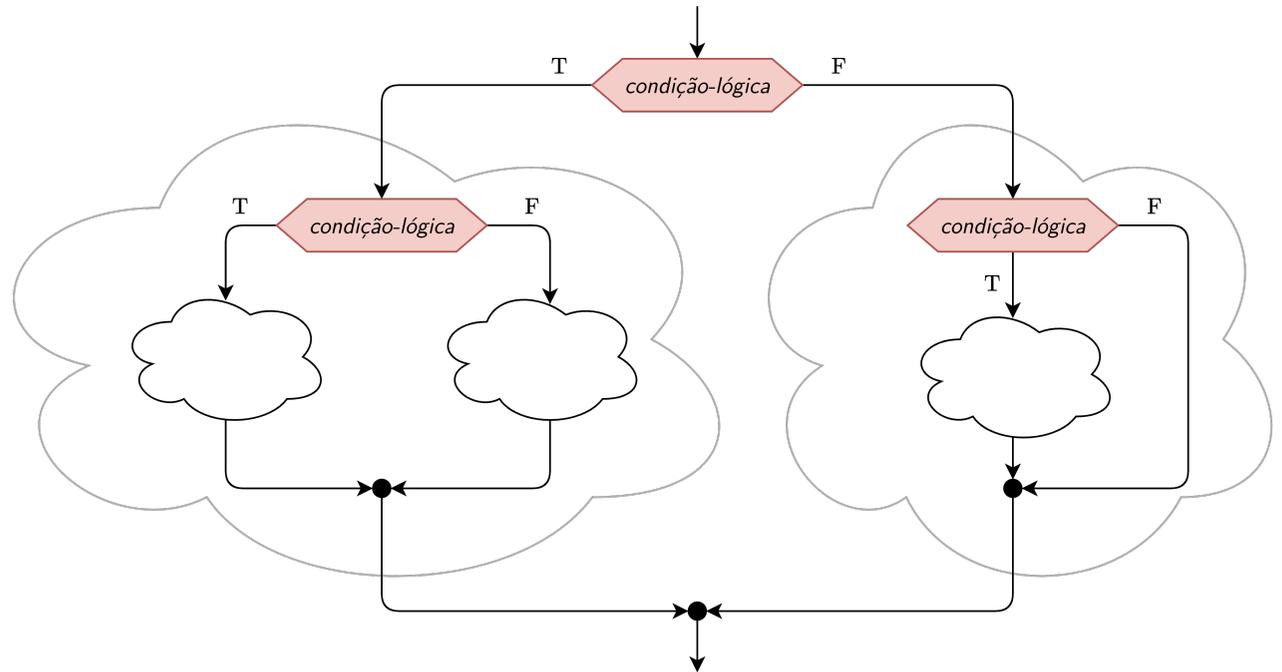
```
1 Início
2 Leia i
3 Leia t
4 Se  $i \ge 65 \vee t \ge 30 \vee (i \ge 60 \wedge t \ge 25)$  então
5     Escreva "Pode"
6 Senão
7     Escreva "Não pode"
8 FimSe
9 Fim
```

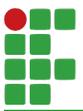


Aninhamento de comandos de decisão

É possível aninhar comandos de decisão, isto é, colocar um comando de decisão dentro de outro.

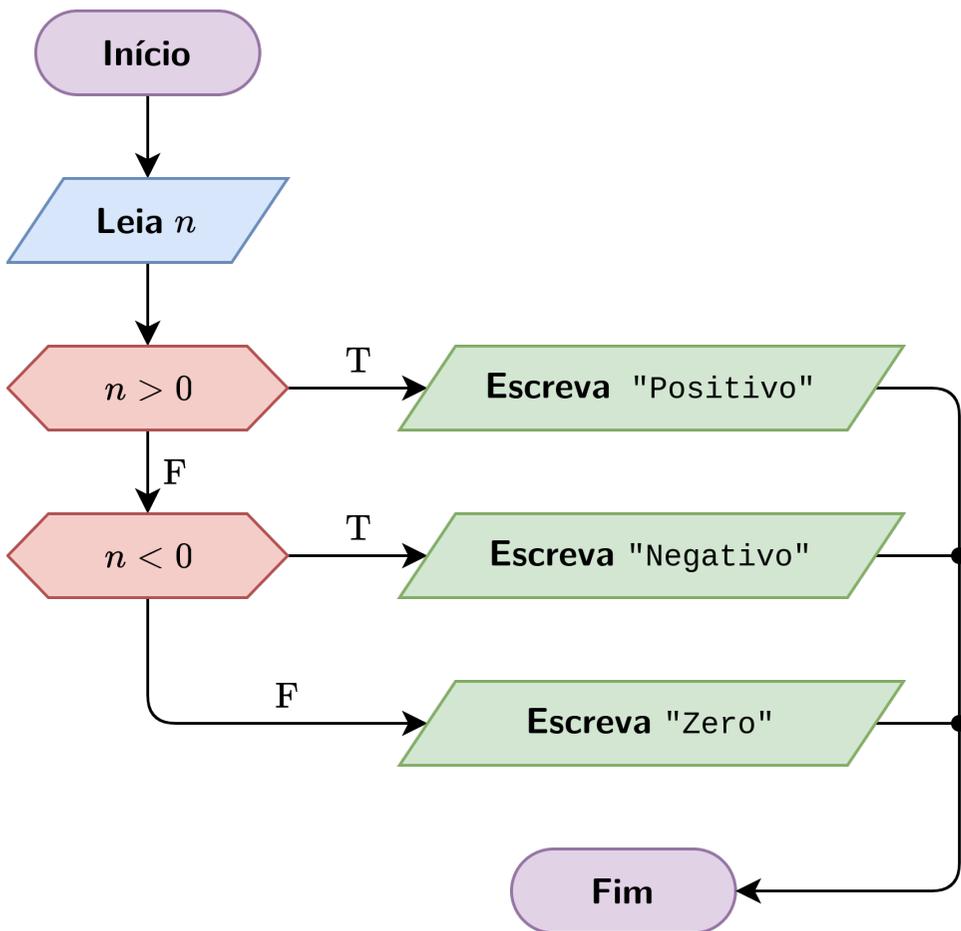
*Exemplo arbitrário de aninhamento de comandos de decisão: Há um comando **Se-Senão**, cujo corpo **Se** contém outro comando **Se-Senão** e corpo **Senão** um comando **Se**.*





Exemplo: Sinal de um número

Elabore um algoritmo que **LÊ** um número inteiro e **ESCREVE** a mensagem **Positivo**, se o número for positivo, **Negativo**, se o número for negativo, ou **Zero**, se o número for igual a zero.



Variáveis

n : inteiro

```
1 Início
2 Leia n
3 Se n > 0 então
4     Escreva "Positivo"
5 Senão
6     Se n < 0 então
7         Escreva "Negativo"
8     Senão
9         Escreva "Zero"
10    FimSe
11 FimSe
12 Fim
```

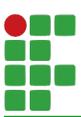


O comando SenãoSe

É possível deixar o pseudocódigo mais legível usando a palavra-chave **SenãoSe**.

```
Se condição-lógica1 então  
    instruções1  
SenãoSe condição-lógica2 então  
    instruções2  
SenãoSe condição-lógica3 então  
    instruções3  
SenãoSe condição-lógica4 então  
    instruções4  
    ⋮  
Senão  
    instruçõesn  
FimSe
```

A sequência de instruções referente a um comando **SenãoSe** será executada quando a condição associada ao comando for verdadeira e *todas* as condições associadas aos comandos **Se** e **SenãoSe** anteriores forem falsas.



Exemplo: Sinal de um número com SenãoSe

Os algoritmos abaixo são completamente equivalentes.

```
1 Início  
2   Leia  $n$   
3   Se  $n > 0$  então  
4     Escreva "Positivo"  
5   Senão  
6     Se  $n < 0$  então  
7       Escreva "Negativo"  
8     Senão  
9       Escreva "Zero"  
10    FimSe  
11   FimSe  
12  Fim
```



```
1 Início  
2   Leia  $n$   
3   Se  $n > 0$  então  
4     Escreva "Positivo"  
5   SenãoSe  $n < 0$  então  
6     Escreva "Negativo"  
7   Senão  
8     Escreva "Zero"  
9   FimSe  
10  Fim
```



Atenção

O algoritmo ao lado funciona, mas **não é equivalente** aos anteriores (é menos eficiente). Por quê?

```
1 Início
2   Leia  $n$ 
3   Se  $n > 0$  então
4     Escreva "Positivo"
5   FimSe
6   Se  $n < 0$  então
7     Escreva "Negativo"
8   FimSe
9   Se  $n = 0$  então
10    Escreva "Zero"
11  FimSe
12 Fim
```



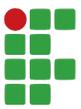
Exercícios em sala de aula

1. Ao comprar ingressos para um festival de música, os fãs têm que pagar uma taxa de conveniência, além do preço do ingresso.
 - Para valores de ingresso de até R\$ 100,00 (inclusive), a taxa de conveniência é de R\$ 8,00 + 5% do valor do ingresso.
 - Para valores superiores a R\$ 100,00, a taxa de conveniência é de R\$ 11,00 + 2% do valor do ingresso.
 - Além disso, o valor mínimo da taxa de conveniência é R\$ 10,00.

Exemplos:

- Um ingresso de R\$ 200,00 resulta em uma taxa de conveniência de R\$ 11,00 + 2% de R\$ 200,00, ou seja, R\$ 15,00.
- Um ingresso de R\$ 10,00 resultaria em uma taxa de R\$ 8,00 + 5% de R\$ 10,00, ou seja, R\$ 8,50; no entanto, o valor mínimo da taxa de conveniência é R\$ 10,00, então a taxa é R\$ 10,00.

Elabore um fluxograma e um pseudocódigo para um algoritmo que **LÊ** o valor do ingresso e **ESCREVE** o valor total a ser pago. Em seguida, efetue três testes de mesa: o primeiro com a entrada **5**, o segundo com a entrada **100** e o terceiro com a entrada **2000**.



2. Elabore um fluxograma e um pseudocódigo para um algoritmo que **LÊ** cinco números reais e **ESCREVE** a média e a quantidade de números (dentre estes cinco números) que estão acima da média. Efetue um teste de mesa com a entrada **3.4, 2.0, 1.5, 4.0, 2.1**; a saída deverá ser **2.6, 2**, pois a média é 2.6 e há dois números acima da média.
3. Elabore um fluxograma e um pseudocódigo para um algoritmo que **LÊ** dois números reais representando o peso e a altura de uma pessoa e **ESCREVE** o **índice de massa corpórea (IMC)** desta pessoa, bem com a **classificação do seu estado nutricional**. Utilize o comando **SenãoSe**. Em seguida, efetue um teste de mesa com a entrada **83, 1.75**; a saída deverá ser **27.102, Sobrepeso**. Veja a [página do Ministério da Saúde](#) para mais informações sobre o IMC e a classificação do estado nutricional.