

ALGORITMO E PROGRAMAÇÃO DE COMPUTADORES: ESTRUTURAS CONDICIONAIS

Matheus Felipe Bühring Spohr

Eduardo Luis Hosda

Faculdade Iguaçu – Capanema – Paraná – Brasil

matheusfelipebuhhringspohr@gmail.com

eduardohosda@gmail.com

Resumo: Este artigo tem como objetivo abordar um dos muitos tópicos de algoritmos e lógica de programação, usufruindo ainda da linguagem do português estruturado, somado a IDE de aprendizagem, Portugol Studio. Terá seu foco nas estruturas condicionais, como elas funcionam e como se comportam.

Abstract: This article aims to address one of the many topics of algorithms and programming logic, also benefiting from the structured Portuguese language, added to Learning IDE, Portugol Studio. It will focus on conditional structures, how they function, and how they behave.

INTRODUÇÃO

Todos aqueles com o mínimo de noção de programação sabe que a lógica é essencial para adentrar neste mundo, e algoritmos exatamente disso. Neste artigo iremos abordar um tema muito utilizado na área de programação, um tópico essencial para se poder programar qualquer software, sem ela não é possível fazer algo que se preze, as estruturas condicionais.

1. Estrutura Condicional

Uma Estrutura Condicional, nos proporciona a escolha de um grupo de atividades quando determinada condição for satisfeita. Em vários momentos da nossa vida, nos deparamos que, para realizar algo, devemos satisfazer uma certa condição. Um ótimo exemplo disso é quando temos que escolher realizar uma tarefa ao invés de outra, por exemplo “Se eu comprar x-burguer então, não terei dinheiro para comer pizza.” e assim por diante. Mas como representar isso em código? Muito simples, utilizamos algo que chamamos de Estrutura condicional.

Estruturas Condicionais, também chamadas de condição, são representadas no português estruturado pelo “se”. Essas condições vão julgar um bloco de comandos, ou seja, vai indicar o bloco de comandos que será executado caso a informação fornecida na estrutura condicional for verdadeira. Caso a condição indicada seja verdadeira, o primeiro bloco de código será executado. Segue o exemplo:

```
1 programa
2 {
3     funcao inicio()
4     {
5         logico Comprar_Hamburguer = verdadeiro
6
7         se(Comprar_Hamburguer)
8         {
9             //Comandos
10        }
11    }
12 }
13
```

Figura 1: Fonte Estrutura Condicional.

Essas estruturas normalmente são separadas por, Estruturas condicionais simples, compostas e encadeadas.

```

1 programa
2 {
3     funcao inicio()
4     {
5         logico condicao = verdadeiro
6         logico condicao2 = verdadeiro
7
8         //Simples:
9         se(condicao){
10            //Comandos
11        }
12
13        //Composta:
14        se(condicao){
15            //Comandos
16        }senao{
17            //Comandos2
18        }
19
20        //Encadeada:
21        se(condicao){
22            //Comandos
23        }senao se(condicao2){
24            //Comandos2
25        }
26    }
27 }
28

```

Figura 2: Estruturas condicionais básicas.

Agora, veremos a seguir cada uma mais detalhadamente:

1.1 Estruturas Condicionais Simples:

A estrutura condicional simples, vai executar somente um bloco de comando se a condição passada na estrutura for verdadeira, caso ela seja falsa, o algoritmo encerra o estrutura condicional e não executará mais nada apenas seguirá em frente ou sua execução será encerrada. Representado em um fluxograma seria da seguinte maneira:

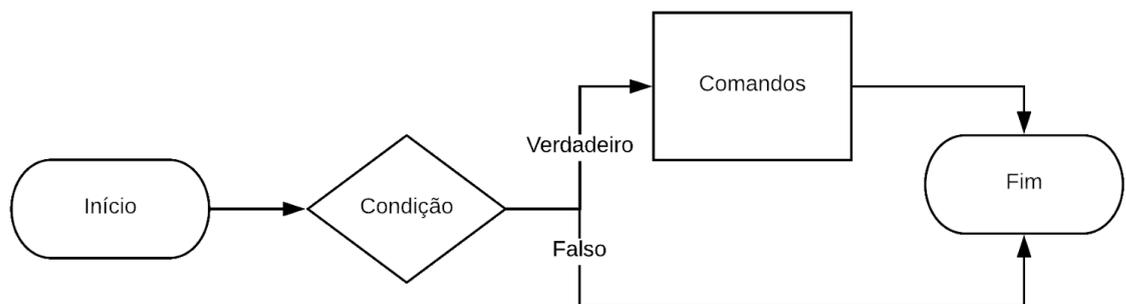


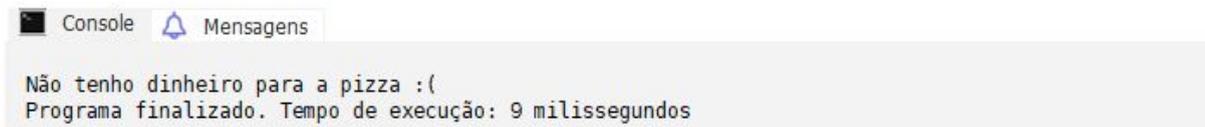
Figura 3: Fluxograma Estrutura Condicional Simples.

Para fazer o uso desta estrutura no Portugol Studio é muito simples, como representado no exemplo abaixo:

```
1 programa
2 {
3     funcao inicio()
4     {
5         logico Comprar_Hamburger = verdadeiro
6
7         se(Comprar_Hamburger){
8             escreva("Não tenho dinheiro para a pizza :(")
9         }
10    }
11 }
12
```

Figura 4: Fonte Estrutura Condicional Simples quando a variável é verdadeira.

Na saída de dados do console será exibido o seguinte resultado:



```
Console Mensagens
Não tenho dinheiro para a pizza :(
Programa finalizado. Tempo de execução: 9 milissegundos
```

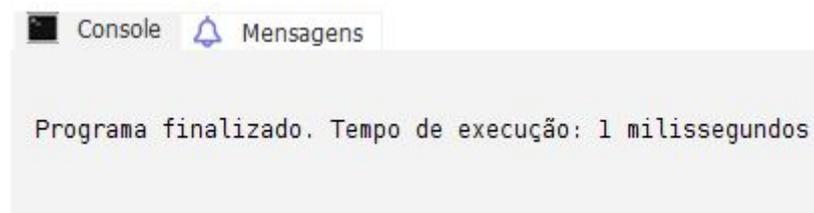
Figura 5: Saída Estrutura Condicional Simples quando a variável é verdadeira

O programa apenas irá mostrar na tela a frase “Não tenho dinheiro para comprar pizza :(“ se, como mostrado a indicação for verdadeira, caso mudássemos o “verdadeiro” para “falso” a escrita na tela já não apareceria mais. Como veremos abaixo:

```
programa
{
    funcao inicio()
    {
        logico Comprar_Hamburger = falso

        se(Comprar_Hamburger){
            escreva("Não tenho dinheiro para pizza :(")
        }
    }
}
```

Figura 6: Fonte Estrutura Condicional Simples quando a variável é falsa.



```
Console Mensagens
Programa finalizado. Tempo de execução: 1 milissegundos
```

Figura 7: Saída Estrutura Condicional Simples quando a variável é falsa.

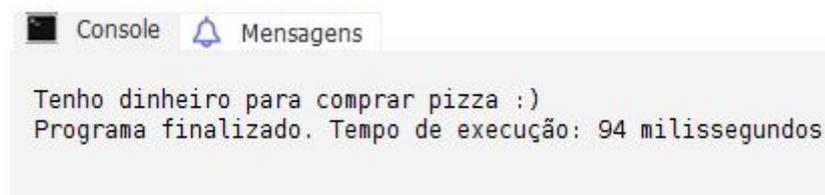
1.2 Estruturas Condicionais Composta:

Esse tipo de estrutura tem o mesmo princípio da anterior, toda a via quando a condição informada no “se” não resulta em verdadeira, o programa parte para um outros comandos caso a informação, ou seja, se a condição for falsa, ele irá executar outro tipo de comando, na sintaxe do português estruturado advindo do portugol studio, utilizamos o “senao”. Conforme mostra a imagem abaixo:

```
programa
{
  funcao inicio()
  {
    logico Comprar_Hamburguer = falso

    se(Comprar_Hamburguer){
      escreva("Não tenho dinheiro para comprar pizza :(")
    }senao{
      escreva("Tenho dinheiro para comprar pizza :)")
    }
  }
}
```

Figura 8: Fonte Estrutura Condicional Composta.



```
Console  Mensagens
Tenho dinheiro para comprar pizza :)
Programa finalizado. Tempo de execução: 94 milissegundos
```

Figura 9: Saída Estrutura Condicional Composta

Caso a variável “Comprar_Hamburguer” fosse verdadeira, neste caso ele retornaria a frase “Não tenho dinheiro para comprar pizza :(“ como mostrado na figura 4.

1.3 Estruturas Condicionais Encadeadas:

Este tipo de estrutura condicional segue o princípio das 2 anteriores, porém há algo que nelas faltam, e se eu quiser fazer mais de uma condição ou verificação? E é exatamente disso que essa estrutura trata.

Como nos exemplos citados acima e se eu não quiser comprar absolutamente nada? Bom para isso iremos utilizar o “senao se” ele irá permitir fazermos mais uma verificação partindo da mesma estrutura, ilimitadamente, ou seja, é permitido realizar quantos “senao se” quiser .

Mas caso eu queira usar só um “se” depois de outro “se”, eu posso? Pode, mas estará fora das boas práticas de programador e corre o risco do “senao” apenas responder a um dos “se” .

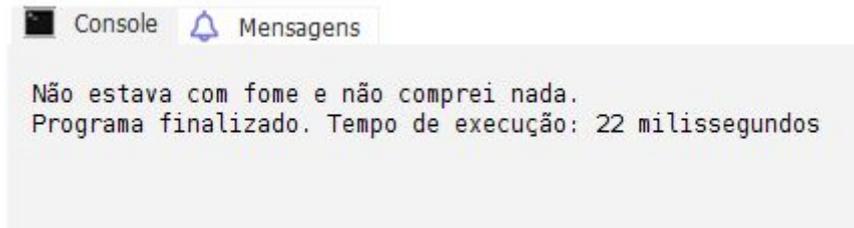
Vejamos abaixo como ela funciona:

```
programa
{
    funcao inicio()
    {
        logico Comprar_Hamburguer = falso
        logico Comprar_Pizza = falso

        se(Comprar_Hamburguer){
            escreva("Não tenho dinheiro para comprar pizza :(")
        }senao se(Comprar_Pizza){
            escreva("Não tenho dinheiro para comer x-burguer :(")
        }senao{
            escreva("Não estava com fome e não comprei nada.")
        }
    }
}
```

Figura 10: Fonte Estrutura Condicional Encadeada quando a variável é “Comprar_pizza” é falsa.

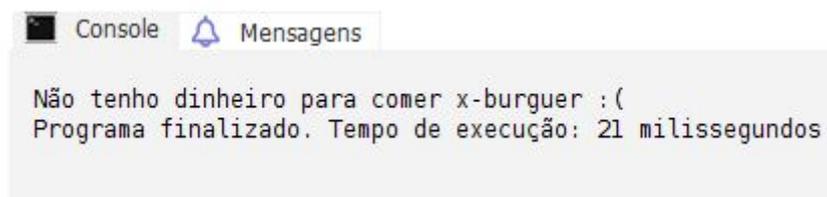
Como mostrado acima, criamos uma nova variável, com o nome de “Comprar_Pizza” e por hora deixaremos-a como falsa, pra ver o que nos retorna, como o demonstrado abaixo:



```
Console Mensagens
Não estava com fome e não comprei nada.
Programa finalizado. Tempo de execução: 22 milissegundos
```

Figura 11: Saída Estrutura Condicional Encadeada quando a variável é “Comprar_pizza” é falsa.

Nos exemplos demonstrados acima o nosso algoritmo não chega a executar o que está dentro do “senao se”, porém se for trocado a atribuição da variável “Comprar_Pizza” para “verdadeiro”, veremos isso acontecer:



```
Console Mensagens
Não tenho dinheiro para comer x-burguer :(
Programa finalizado. Tempo de execução: 21 milissegundos
```

Figura 12: Saída Estrutura Condicional Encadeada quando a variável é “Comprar_pizza” é verdadeira.

Conclusão

Desta forma, torna-se notável a importância da existência de estruturas condicionais. Estas estruturas possuem as mais diversas e inimagináveis utilidades dentro de uma determinada aplicação e, é por isso que são tão relevantes para um programador, não há como criar um software sem a existência delas. Por isso, foi buscado explorar da maneira mais didática possível, como são e para que serve.

Bibliografia

Algoritmos e Programação De Computadores. Tradução . [s.l.] Elsevier Editora Ltda., 2012.

Condicionais Simples. Disponível em:

<<https://quetalumprograma.wordpress.com/estrutura-condicional/simples>
>

LAGE, B. **Aprendendo a Programar em Python: Estruturas Condicionais - IF.** Disponível em:

<<https://www.devmedia.com.br/aprendendo-a-programar-em-python-estruturas-condicionais-if/17358>>

SILVA, C. S. D. A. **Fábrica de Software » Algoritmo – Estruturas Condicionais.** Disponível em:

<<http://fabrica.ms.senac.br/2013/06/algoritmo-estruturas-condicionais/>>

SOUZA, M. A. F. D. E. **Algoritmos e lógica de programação.** Tradução . [s.l.] Thomson Learning, 2006.

ST, L. G.; SOUZA. **Estruturas condicionais em Java.** Disponível em:

<<https://www.devmedia.com.br/estruturas-condicionais-em-java/21135>>