

# Lógica de Programação

## Aula 08 - Estruturas Aninhadas: ENQUANTO e PARA



## Apresentação da Aula

---

Nas aulas anteriores, você conheceu as estruturas de repetição **COM** variável de controle e **SEM** variável de controle. Nesta aula, você conhecerá o **aninhamento** das estruturas de repetição **enquanto** e **para**. É muito importante que você esteja bem atento a essas estruturas, pois elas serão utilizadas na maioria dos algoritmos construídos. Vamos conferir? 😏 (cara piscando - winking face.)



### Objetivos

Conhecer de que forma acontece o **aninhamento** das estruturas de repetição **enquanto** e **para**.

Exercitar a utilização das estruturas de repetição **enquanto** e **para**.

# Aninhamento de Estruturas de Repetição

---

Em aulas anteriores, você conheceu a estrutura de decisão **se...então...senão**. Viu que, para construir alguns fluxos de decisão, é necessário utilizar mais de uma estrutura de decisão e, em determinados casos, é preciso, também, utilizar as estruturas de decisão de forma aninhada, ou seja, uma interna a outra.

Nesta aula, você verá a sintaxe relativa ao **aninhamento** das estruturas de repetição vistas anteriormente. Esse processo será importante, principalmente, quando for necessário trabalhar com as estruturas de dados homogêneas que você estudará nas aulas seguintes.

Você já conheceu as seguintes estruturas de repetição: a estrutura **COM** variável de controle **para** e a estrutura **SEM** variável de controle **enquanto**. Viu, ainda, quais as situações apropriadas para se utilizar cada uma dessas estruturas. Além disso, aprendeu que é possível utilizar qualquer uma dessas estruturas na maioria dos problemas que envolvam o uso de uma estrutura de repetição.

Assim, trabalharemos, primeiro, com o **aninhamento** da estrutura de repetição **enquanto** e, em seguida, com o **aninhamento** da estrutura **para**.

## Aninhamento de 'enquanto'

---

Lembrando brevemente o que você viu anteriormente sobre a estrutura de repetição **enquanto**: essa é uma estrutura **sem a variável de controle**, portanto, é a estrutura mais apropriada quando não se conhece ou não se sabe o número de repetições (laços) que serão executados.

Nesse sentido, é importante sempre definir a expressão que será utilizada no teste lógico dessa estrutura. Você deve sempre ficar atento e tomar o cuidado para que a expressão permita a conclusão da quantidade de repetições de forma apropriada e, assim, não deixar que a estrutura permaneça repetindo indefinidamente, o que acarretará o chamado **Loop Infinito**.

Relembre agora a sintaxe dessa estrutura:

```
1 enquanto <condição> faça
2   <instruções>
3 fim
```

Agora que você já está com a memória refrescada no que diz respeito à estrutura de repetição **enquanto**, conhecerá a sintaxe de como realizar o **aninhamento** dela.

Observe o código abaixo.

```
1 enquanto <condição> faça
2   <instruções no enquanto externo e executadas em quantidade igual ao número de repetições do
3
4   enquanto <condição> faça
5     <instruções no enquanto aninhado (interno) e executadas em quantidade igual ao número de
6     interno multiplicada pela quantidade de repetições do enquanto externo>
7   fim
8 fim
9
```

No exemplo, você observou a presença das duas estruturas de repetição, uma interna a outra? A estrutura interna possui a mesma sintaxe da estrutura externa, porém, todo o seu código está na área de comandos do **enquanto** externo.

Veja, agora, o exemplo com um pequeno código, em **Potigol**. Observe que nas linhas 1 e 2 foram declaradas duas variáveis *i* e *j*, e ambas possuem o valor inicial igual a 1. Na linha 3, inicia-se a primeira estrutura de repetição **enquanto**, que é encerrada na linha 16. Aninhada a essa estrutura, há uma outra repetição **enquanto**, que começa na linha 8 e termina na linha 11.

Execute o código e observe o resultado. Em seguida, analise o código com atenção, buscando compreender o seu funcionamento. Sinta-se à vontade para modificar os valores de parada das expressões presentes em cada uma das estruturas de repetição e, posteriormente, avalie o resultado gerado pelo algoritmo.

```
1 var i := 1
2 var j := 1
3 enquanto i <= 5 faça
4     escreva " i = {}"
5     i := i + 1
6
7     j := 1
8     enquanto j <= 10 faça
9         imprima " j = {}"
10        j := j + 1
11    fim
12
13    escreva "" # Inserindo uma quebra de linha
14    escreva "" # Inserindo outra quebra de linha
15
16 fim
17
```



E aí? O que você conseguiu perceber nesse código?

Notou que a variável **i** foi impressa apenas cinco vezes, mas a variável **j** foi impressa 50 vezes?

Isso aconteceu porque o **j** foi impresso desde o valor 1 até o número 10, mas esse processo ocorreu 5 vezes. Cada vez que o valor de **j** alcançou o valor máximo e encerrou a execução da estrutura **enquanto** interna, o valor **i** foi incrementado em apenas 1. Por esse motivo, o valor de **j** é impresso 50 vezes, isto é,  $10 \times 5$ .

Em outras palavras, para que ocorra um ciclo (laço) completo da estrutura de repetição **externa**, todo o código interno desta precisa ser executado. Porém, quando se há uma estrutura de repetição **interna** sendo aninhada, é necessário que a estrutura **interna** seja processada completamente, concluindo todos os seus laços, para que, então, as linhas seguintes de código, após a estrutura **interna**, possam ser executadas e, assim, concluir o laço da estrutura **externa**.

# Aninhamento de 'para'

---

Na aula sobre a estrutura de repetição **para**, você deve ter percebido que, em todos os exemplos, utilizei como variável de controle a letra 'i'. Como dito, a utilização dessa letra como variável de controle para as estruturas de repetição **para** é muito comum, porém, isso não é uma convenção. Portanto, lembre-se de que, se preferir, poderá definir outro nome para a variável de controle e tornar ainda mais fácil a sua leitura do algoritmo.



## Atenção

Vale lembrar que a definição do nome da variável de controle deve seguir as mesmas regras da definição de qualquer variável.

A estrutura de repetição **para interna** deve possuir o nome da variável de controle diferente da variável de controle do **para externo**. Por esse motivo, a definição do nome da variável de controle em uma estrutura de repetição **para** torna-se importante, principalmente, quando é necessário realizar a construção de algoritmos com o **aninhamento** de duas ou mais dessa estrutura.

Em outras palavras, no **aninhamento** de **para** não deve ser utilizado o mesmo nome da variável de controle em mais de uma estrutura, caso contrário, o comportamento de **incremento** da variável de controle ocorrerá de forma desordenada.

No **aninhamento** da estrutura de repetição **para**, você terá a execução de um laço do **para externo** somente após a execução de todos os laços do **para interno**. Assim, somente após a conclusão de todos os laços da estrutura de repetição **para interno** é que é realizado um laço na estrutura **externa**.

## #FicaDica

Isso te lembra algo? Esse comportamento é semelhante ao das estruturas de repetição **enquanto**.

No exemplo abaixo, apresento uma pequena tabuada para ilustrar como isso ocorre. Observe com atenção o conteúdo impresso nesse algoritmo e, em seguida, procure modificar os valores iniciais e finais de cada um dos **para** presentes no exemplo, mas fique atento em como isso interfere no resultado impresso.

Observe, principalmente, quais são os resultados obtidos a partir da mudança de cada uma das estruturas. Sugiro que você altere somente um dos elementos por vez, ou seja, experimente alterar, por exemplo, o valor 3 do **para interno**, definindo um outro valor de sua escolha. Verifique o resultado e, só então, após realizar esse experimento, você alterará algum dos outros parâmetros, sempre analisando qual é o impacto no resultado gerado.

```
1 # Exemplo com aninhamento de para
2 para multiplicador de 1 até 4 faça
3   para multiplicando de 1 até 3 faça      # escreve a tabuada {1..4} x {1..3}
4     escreva "{multiplicador} * {multiplicando} = {multiplicador * multiplicando}"
5   fim
6 fim
7
```

Caso você ainda tenha dúvidas sobre como aninhar as estruturas de repetição **para**, procure o seu professor mediador para que ele possa lhe auxiliar com esse exemplo.



## É possível aninhar a estrutura 'para' dentro do 'enquanto', e vice-versa?

Sim, é possível aninhar qualquer das estruturas de repetição uma dentro da outra, mas sempre obedecendo à sintaxe de cada uma. Assim, você pode ter uma estrutura **enquanto** dentro da estrutura **para**, como também você pode ter a estrutura **para** dentro da estrutura **enquanto**. Apenas para exemplificar essa possibilidade, apresento abaixo o código que calcula a tabuada utilizando ambas as estruturas de forma aninhada.

```
1 var multiplicando := 0
2 para multiplicador de 1 até 4 faça
3   multiplicando := 1
4   enquanto multiplicando <= 3 faça
5     escreva "{multiplicador} * {multiplicando} = {multiplicador * multiplicando}"
6     multiplicando := multiplicando + 1
7   fim
8 fim
9
```



### Atividade 01

#### Enquanto...

Utilizando a estrutura **enquanto**, construa um algoritmo que recebe 2 números inteiros (X e Y) e apresenta uma matriz X por Y.

Por exemplo, para os inteiros 3 e 3, a matriz impressa é:

```
1.1 1.2 1.3
2.1 2.2 2.3
3.1 3.2 3.3
```

Obs.1: Há um espaço em branco ao final de cada linha da matriz.

Obs.2: Há uma quebra de linha após a impressão do último termo da matriz.





## Atividade 02

### Vai de para!

Utilizando a estrutura **para**, construa um algoritmo que recebe 2 números inteiros (X e Y) e apresenta uma matriz X por Y.

Por exemplo, para os inteiros 3 e 3, a matriz impressa é:

```
1.1 1.2 1.3
2.1 2.2 2.3
3.1 3.2 3.3
```

Obs.1: Há um espaço em branco ao final de cada linha da matriz.

Obs.2: Há uma quebra de linha após a impressão do último termo da matriz.

## Quando e por que utilizar o Aninhamento?

A utilização das estruturas de repetição de forma aninhada não possui uma regra específica que determine exatamente em quais situações é preciso aninhá-las.

Por esse motivo, você deverá analisar cada problema e, a partir do que se é solicitado, determinar se a solução utilizará estrutura de repetição, avaliando, também, se há necessidade do **aninhamento** de uma ou de mais estruturas de repetição.

Espero que você tenha realizado todos os exercícios propostos nesta aula e compreendido os conteúdos relativos ao **aninhamento** das estruturas de repetição. Caso algo ainda não tenha ficado claro, procure modificar os exemplos dos códigos propostos e observar o impacto causado no algoritmo apresentado; isso certamente deve lhe auxiliar a compreender melhor o assunto estudado.

Até a próxima aula! 😊 (cara piscando - winking face.)



## Resumo

---

Nesta aula, você aprendeu como é realizado o **aninhamento** das estruturas de repetição. Viu, também, que é possível realizar esse processo utilizando ambas as estruturas simultaneamente (**para** e **enquanto** em um mesmo algoritmo).

O uso aninhado de estrutura de repetição é de grande importância quando é necessário trabalhar com estruturas de dados homogêneas bidimensionais, as quais você conhecerá nas próximas aulas.



## Referências

---

- **Linguagem Potigol: Programação para todos.** Disponível em: <http://potigol.github.io/>. Acesso em: 29 ago. 2018.