

Aula03 - Exercícios de fixação

Davi Trost Gouveia - PC3010741

Exercício 1

```
1. public static void func(int n){  
2.     for(int i=0; i<n; i++){  
3.         print(i);  
4.     }  
5.     for(int i=0; i<n; i++){  
6.         print(i);  
7.     }  
8. }
```

for Loop

Inicialização:

- Atribuição: (int i=0;) = 1
- Comparação: (i<n;) = 1

Execução:

- Comparação: (i<n;) = n
- Incremento : (i++) = n

Execução do comando:

- Execução do print: n

Primeiro loop tem um total de: $3n + 2$ instruções

Como os dois loops não estão aninhados, basta multiplicar a contagem por 2.

Portanto, o algoritmo possui **$6n + 4$** instruções.

Plotando o gráfico

Valor de N x Contagem de instruções

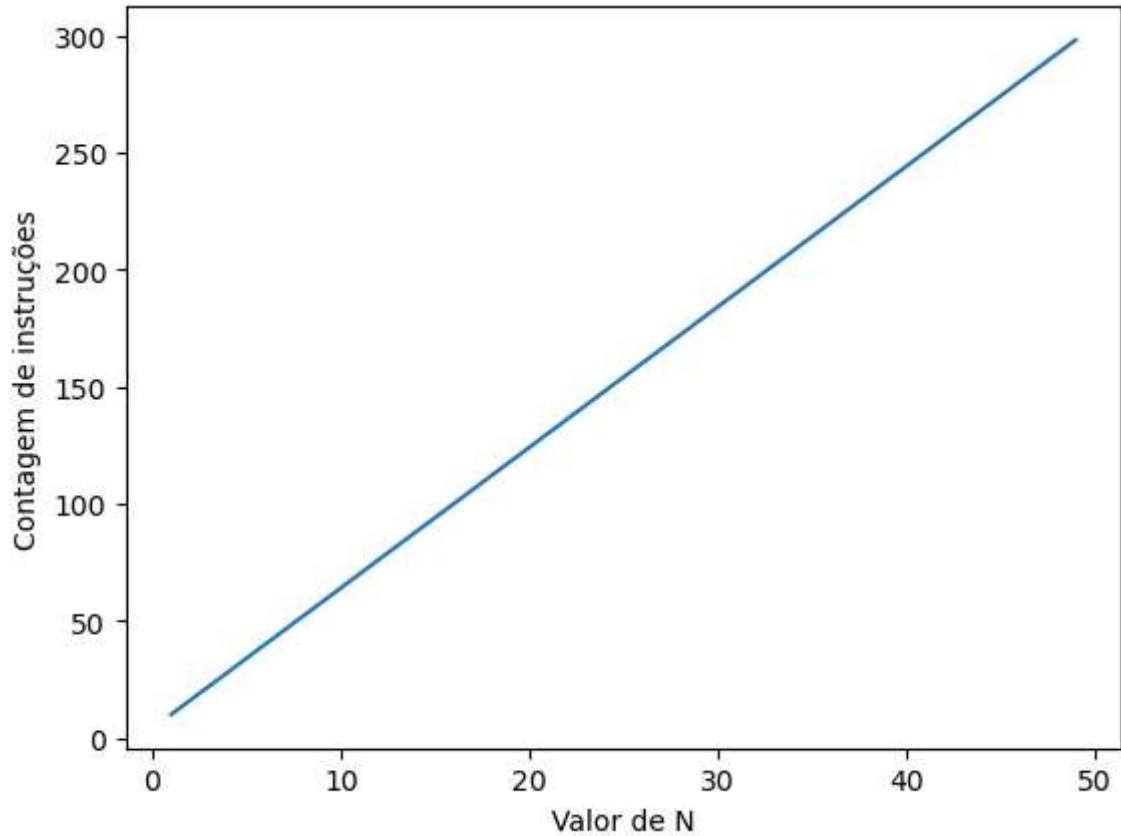
```
In [1]: import matplotlib.pyplot as plt
```

```
In [2]: def countInstructions(n):  
         instructionsTotal = 6*n + 4  
         return instructionsTotal
```

```
In [3]: n_count = range(1,50)

instructions_count = [countInstructions(n) for n in n_count]

plt.plot(n_count, instructions_count)
plt.xlabel('Valor de N')
plt.ylabel('Contagem de instruções')
plt.show()
```



Exercício 2

```
1. public static void func(int n){
2.     for(int i=0; i<n/2; i++){
3.         print(i);
4.     }
5. }
```

Inicialização:

- Atribuição: (int i=0;) = 1
- Comparação: (i<n/2;) = 1

Execução:

- Comparação: (i<n;) = $n/2$
- Incremento : (i++) = $n/2$

Execução do comando:

- Execução do print: $n/2$

O algoritmo possui $3n/2 + 2$ instruções.

Plotando o gráfico

Valor de N x Contagem de instruções

```
In [4]: def countInstructions(n):  
        instructionsTotal = 3*n/2 + 2  
        return instructionsTotal
```

```
In [5]: n_count = range(1,50)  
  
        instructions_count = [countInstructions(n) for n in n_count]  
  
        plt.plot(n_count, instructions_count)  
        plt.xlabel('Valor de N')  
        plt.ylabel('Contagem de instruções')  
        plt.show()
```

