



# Missão

## Módulo 5 - Python

*Preâmbulo: Neste Módulo, vemos como usar arrays e suas funções associadas.*

*Versão: 3.0*

# Sumário

I	Uma palavra sobre esta Missão	2
II	Introdução	3
III	Instruções Gerais	4
IV	Exercício 00: <code>create__array</code>	5
V	Exercício 01: <code>play__with__arrays</code>	6
VI	Exercício 02: <code>play__with__arrays++</code>	7
VII	Exercício 03: <code>play__with__arrays+=2</code>	8
VIII	Exercício 04: Parâmetros	9
IX	Entrega e avaliação por pares	10

# Capítulo I

## Uma palavra sobre esta Missão

Bem-vindo(a)!

Você começará um Módulo desta Missão de programação de computadores. Nosso objetivo é apresentá-lo(a) ao código por trás do software que você usa diariamente e imergir na aprendizagem entre pares, o modelo educacional da 42.

Programação é sobre lógica, não matemática. Ela lhe dá blocos de construção básicos que você pode montar de inúmeras maneiras. Não existe uma única solução “correta” para um problema—sua solução será única, assim como as soluções de cada um dos seus colegas.

Rápido ou lento, elegante ou bagunçado, contanto que funcione, é o que importa! Esses blocos de construção formarão uma sequência de instruções (para cálculos, exibições, etc.) que o computador executará na ordem que você projetar.

Em vez de fornecer um curso onde cada problema tem apenas uma solução, nós o(a) colocamos em um ambiente de aprendizado entre pares. Você pesquisará elementos que possam ajudá-lo(a) a enfrentar seu desafio, refiná-los por meio de testes e experimentação e, finalmente, criar seu próprio programa. Discuta com outras pessoas, compartilhe suas perspectivas, apresente novas ideias juntos e teste tudo sozinho(a) para garantir que funcione.

A avaliação por pares é uma oportunidade fundamental para descobrir abordagens alternativas e detectar possíveis problemas em seu programa que você pode ter perdido (considere como uma falha de programa pode ser frustrante). Cada revisor(a) abordará seu trabalho de forma diferente—como clientes com diferentes expectativas—dando a você novas perspectivas. Você pode até formar conexões para futuras colaborações.

Ao final desta Missão, sua jornada será única. Você terá enfrentado diferentes desafios, validado diferentes projetos e escolhido caminhos diferentes dos outros—e isso é perfeitamente bom! Esta é uma experiência coletiva e individual, e todos ganharão algo com isso.

Boa sorte a todos; esperamos que você aproveite esta jornada de descoberta.

# Capítulo II

## Introdução

O que este Módulo mostrará a você:

- Você aprenderá como lidar com arrays e suas funções associadas.

# Capítulo III


## Instruções Gerais

A menos que especificado de outra forma, as seguintes regras se aplicam todos os dias desta Missão.

- Este documento é a única fonte confiável. Não confie em boatos.
- Este documento pode ser atualizado até uma hora antes do prazo de envio.
- As tarefas devem ser concluídas na ordem especificada. As tarefas posteriores não serão avaliadas, a menos que todas as anteriores sejam concluídas corretamente.
- Preste muita atenção aos direitos de acesso de seus arquivos e pastas.
- Suas tarefas serão avaliadas por seus colegas Missão.
- Todas as tarefas de shell devem ser executadas usando `/bin/bash`.
- Você não deve deixar nenhum arquivo em seu espaço de trabalho de envio além daqueles explicitamente solicitados pelas tarefas.
- Tem uma pergunta? Pergunte ao seu vizinho à sua esquerda. Caso contrário, tente o seu vizinho à sua direita.
- Cada resposta técnica que você precisa pode ser encontrada nas páginas `man` ou online.
- Lembre-se de usar o servidor do Discord!
- Leia os exemplos cuidadosamente, pois eles podem revelar requisitos que não são imediatamente óbvios na atribuição descrição.
- Por Thor, por Odin! Use seu cérebro!!!

# Capítulo IV

## Exercício 00: create\_array


	Exercício : 00
Criar um array	
Pasta de entrega : <i>ex00/</i>	
Arquivos para entregar : <b>create_array.py</b>	
Funções ou bibliotecas autorizadas : Todas	

- Crie um programa chamado `create_array.py`.
- Garanta que este programa seja executável.
- Defina um array de números.
- Exiba o array na tela:

```
?> ./create_array.py | cat -e
[2, 8, 9, 48, 8, 22, -12, 2]$
?>
```

# Capítulo V

## Exercício 01: play\_with\_arrays


	Exercício : 01
Manipulando arrays	
Pasta de entrega : <i>ex01/</i>	
Arquivos para entregar : <code>play_with_arrays.py</code>	
Funções ou bibliotecas autorizadas : Todas	

- Crie um programa chamado `play_with_arrays.py`.
- Garanta que este programa seja executável.
- Defina um array de números.
- Itere sobre este array, criando um novo array adicionando 2 a cada valor no array original.
- Seu programa deve conter dois arrays: o array original e o array modificado.
- Exiba ambos os arrays na tela.
- Por exemplo, se o array original for `[2, 8, 9, 48, 8, 22, -12, 2]`, a saída deve ser:

```
?> ./play_with_arrays.py
Original array: [2, 8, 9, 48, 8, 22, -12, 2]
New array: [4, 10, 11, 50, 10, 24, -10, 4]
?>
```

# Capítulo VI

## Exercício 02: `play_with_arrays++`

	Exercício : 02
Manipulando arrays, mais uma vez	
Pasta de entrega : <code>ex02/</code>	
Arquivos para entregar : <code>play_with_arrays.py</code>	
Funções ou bibliotecas autorizadas : Todas	


- Modifique seu programa anterior para processar apenas os valores maiores que 5 no array original.
- Por exemplo, se seu array original for `[2, 8, 9, 48, 8, 22, -12, 2]`, a saída deve ser:

```
?> ./play_with_arrays.py | cat -e
[2, 8, 9, 48, 8, 22, -12, 2]$
[10, 11, 50, 10, 24]$
?>
```



# Capítulo VII

## Exercício 03: `play_with_arrays` += 2

	Exercício : 03
Manipulando Arrays, Ainda!	
Pasta de entrega : <code>ex03/</code>	
Arquivos para entregar : <code>play_with_arrays.py</code>	
Funções ou bibliotecas autorizadas : Todas	

- Modifique seu programa anterior para remover duplicatas na saída. Você não deve remover explicitamente valores de seus arrays.
- Para este exercício, o uso de **Set** é obrigatório.
- Por exemplo, se seu array original for `[2, 8, 9, 48, 8, 22, -12, 2]`, a saída deve ser semelhante a esta:


```
?> ./play_with_arrays.py | cat -e
[2, 8, 9, 48, 8, 22, -12, 2]$
{10, 11, 50, 24}$
?>
```



Pesquise sobre conjuntos (sets) em Python.

# Capítulo VIII

## Exercício 04: Parâmetros

	Exercício : 04
	Sim, mas quantos?
	Pasta de entrega : <i>ex04/</i>
	Arquivos para entregar : <b>parameters.py</b>
	Funções ou bibliotecas autorizadas : Todas

- Crie um programa chamado `parameters.py`.
- Garanta que este programa seja executável.
- O programa deve exibir o número de parâmetros passados para ele, seguido por uma nova linha.

```
?> ./parameters.py
Número de parâmetros: 0.
?> ./parameters.py "iniciação"
Número de parâmetros: 1.
?> ./parameters.py "isso" "é" "louco" "tem" "em" "todo" "lugar!"
Número de parâmetros: 7.
?>
```



Consulte a lista `sys.argv` e a função `len()`.

# Capítulo IX

## Entrega e avaliação por pares

- Você deve ter a pasta `missao` na raiz do seu diretório pessoal.
- Dentro da pasta `missao`, você deve ter uma pasta chamada `modulo5`.
- Dentro da pasta `modulo5`, você deve ter uma pasta para cada exercício.
- O Exercício 00 deve estar na pasta `ex00`, o Exercício 01 na pasta `ex01`, etc.
- Cada pasta de exercício deve conter os arquivos solicitados na tarefa.



Observe que, durante sua defesa, qualquer coisa que não esteja presente na pasta do módulo não será verificado.