



**Data Science
Academy**

www.datascienceacademy.com.br

Introdução à Lógica de Programação

Este é um curso expresso e gratuito de Introdução à Lógica de Programação, totalmente dinâmico e objetivo, para ajuda-lo a acompanhar as aulas nos demais cursos na DSA! Aproveite!

E sem perder tempo, já vamos começar com a primeira aula!

A programação de computadores em geral causa certo medo dos alunos. Para falar a verdade causa pânico mesmo (o pânico só não é maior que aquele causado pelo idioma inglês). Pela nossa experiência aqui na DSA, o medo se deve a 2 fatores:

1- O aluno tenta decorar comandos e palavras-chave. É claro que não vai conseguir e isso causa aversão à programação.

2- O aluno não compreende que programar é na verdade desenvolver o raciocínio lógico, conforme demonstrarei logo abaixo, e traduzir esse raciocínio em uma sequência de operações que serão executadas pelo computador.

Passe a olhar para a programação de forma diferente e tudo vai fazer mais sentido! Vamos treinar!

Você já ouviu falar na Sequência de Fibonacci?

Na Matemática, a Sucessão de Fibonacci (também conhecida como Sequência de Fibonacci), é uma sequência de números inteiros, começando normalmente por 0 e 1, na qual, cada termo subsequente corresponde à soma dos dois anteriores.

A sequência recebeu o nome do matemático italiano Leonardo de Pisa, mais conhecido por Fibonacci, que descreveu, no ano de 1202, o crescimento de uma população de coelhos, a partir desta sequência. Esta sequência já era, no entanto, conhecida na antiguidade.

Os números de Fibonacci são, portanto, os números que compõem a seguinte sequência:

0,1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, ...

Em termos matemáticos, a sequência é definida recursivamente pela fórmula abaixo:

$$F_n = F_{n-1} + F_{n-2}$$

A sequência de Fibonacci tem aplicações na análise de mercados financeiros, na ciência da computação e na teoria dos jogos. Quem assistiu ao filme “O Código Da Vinci” deve se lembrar que o Código começa com a Sequência de Fibonacci.

Depois dessa introdução, tenho um problema no qual precisaremos da ajuda da programação para resolver. Aqui está o problema:

Crie um algoritmo em Python que encontre o valor em qualquer posição na Sequência de Fibonacci!

Como resolvemos esse problema via programação? Os mais afobados vão tentar buscar as palavras-chave e começar a criar o programa. Aí está o equívoco. O ideal, principalmente para quem está começando, é criar o algoritmo no formato de pseudo-código, uma sequência de atividades que devem ser realizadas para resolver problema e somente depois traduzimos o pseudo-código em um programa de computador. O que acha dessa solução abaixo?

Vamos definir n como sendo o valor numérico que representa a posição de um número na Sequência de Fibonacci! Leia de novo a frase: Vamos definir n como sendo o valor numérico que representa a posição de um número na Sequência de Fibonacci!

Nosso pseudo-código (levemente customizado) seria assim:

- 1- Defina uma variável n que representa a posição do número na Sequência de Fibonacci!
- 2- Se n for menor que 0, será inválido, pois a Sequência de Fibonacci não tem valor em posição negativa.
- 3- Se n não for menor que 0, verificamos se n é igual a 1. Se a posição for igual a 1 retornamos 0, pois esse é o primeiro valor na Sequência de Fibonacci!
- 4- Se n não for igual a 1, verificamos se n é igual a 2. Se a posição for igual a 2 retornamos 1, pois esse é o segundo valor na Sequência de Fibonacci!
- 5- Se n não estiver em nenhuma das condições anteriores, podemos então encontrar o valor da posição n aplicando a regra: $F_n = F_{n-1} + F_{n-2}$
- 6- Como no passo 5 teremos que calcular todos os números da Sequência de Fibonacci até a posição n , teremos que aplicar recursividade.

Pronto, problema resolvido! Não escrevemos uma única linha de código ainda. Apenas definimos uma sequência lógica de operações para resolver o problema. Agora sim, podemos traduzir essa sequência para uma linguagem de programação, em nosso caso Python. Vamos criar uma função para essa tarefa:

Obs: Os traços abaixo são apenas para representar a indentação, que em Python faz parte da sintaxe. Ao criar a função remova os traços e use espaços ou tabs, conforme você aprendeu no curso gratuito de Python Fundamentos.



Função Python para encontrar o valor em qualquer posição na Sequência de Fibonacci

```
def encontraFibonacci(n):
----if n < 0:
-----print("Entrada incorreta!")
----elif n == 1:
-----return 0
----elif n == 2:
-----return 1
----else:
-----return encontraFibonacci(n-1) + encontraFibonacci(n-2)
```

Perceba que na última linha usamos a recursividade, fazendo uma chamada à própria função. Esse é um recurso muito usado em programação.

Considerando $n = 5$, analise o código acima e busque compreender porque a recursividade está sendo usada! Usamos recursividade para encontrar os valores nas posições de 1 a 4 e assim encontrar o valor na posição 5. Com a recursividade, a função ficará sendo executada em loop até chegar ao resultado final.

Recursividade não é a única forma de resolver esse problema. Poderíamos usar Programação Dinâmica ou Otimização Espacial, conforme mostrarei nas próximas aulas. Essas técnicas de programação são amplamente usadas em diversos algoritmos de Machine Learning.

Criada a função no shell da linguagem Python (ou via Jupyter Notebook), você pode testar assim:

```
# Executa a função para a posição 5
print(encontraFibonacci(5))
```

>> Resultado: 3

```
# Executa a função para a posição 21
print(encontraFibonacci(21))
```

>> Resultado: 6765

```
# Executa a função para a posição 32
print(encontraFibonacci(32))
```

>> Resultado: 1346269

Nenhum conhecimento avançado em Python foi necessário. Diante de um problema, criamos uma sequência lógica de operações e depois convertemos para uma linguagem de programação. Perceba que os comandos e palavras-chave usados na função Python são super simples. O maior desafio foi mesmo na lógica. E para isso, tudo que você precisa é desenvolver seu raciocínio lógico.

Essa aula foi apenas o aperitivo. Em breve começamos a servir o prato principal!

#aula01