

Programação Aplicada Ao Direito



UFMG



MARCELO CANDIOTTO
Sociedade de Advogados MCSA



ASSOCIAÇÃO
BRASILEIRA DE
LAWTECHS &
LEGALTECHS

Conteúdo

Dia 03 - Algoritmos e Estruturas Lógicas

- Algoritmos
- Estruturas Lógicas
- Estruturas de Repetição
- Exercícios Práticos: Análise de um Documento Jurídico, suas variáveis e sua estrutura lógica

Algoritmos

Lógica de programação

Podemos definir “lógica de programação” como a elaboração de sequências de ações para atingir um determinado objetivo (algoritmos). O processo envolve o uso de dispositivos lógicos, como estruturas condicionais (if/else) e de repetição (for/while).

Na computação, as sequências são escritas utilizando as linguagens de programação, que são os “idiomas” compreendidos pelas máquinas.

Algoritmos

Iremos utilizar a linguagem **Python** nesse curso.





Atenção



Algoritmo

!=

Logaritmo

!=

Algarismo

!=

Algorítimo



Selo em Homenagem a Mohamed ibn Musa al-Khwarizmi

Algoritmos

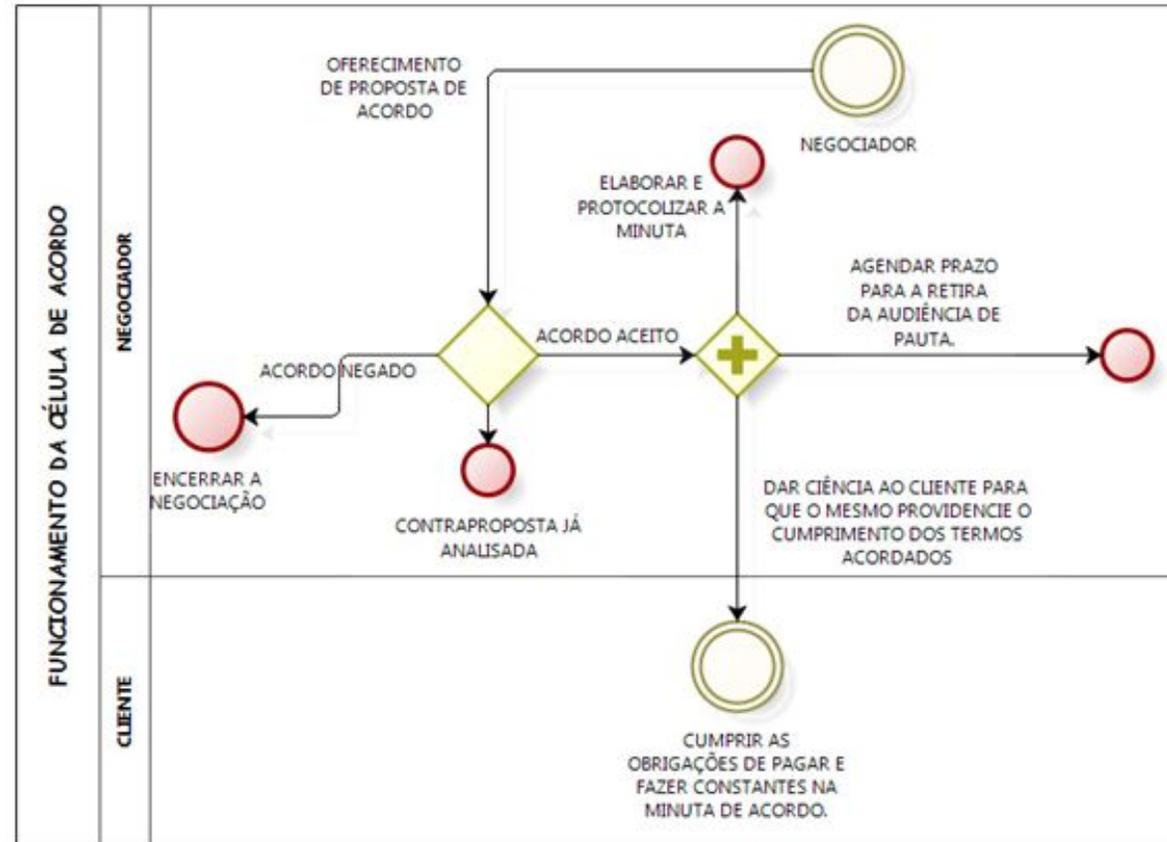
O que é um algoritmo?

É qualquer sequência finita de passos que levam à execução de uma certa tarefa ou à resolução de um problema. Isso é aplicável em vários âmbitos do dia a dia. Veja o exemplo de uma sequência de ações que poderiam ser seguidas por qualquer pessoa que precisa ir ao trabalho todas as manhãs:

1. acordar às 07:00
2. levantar-se
3. tomar café
4. escovar os dentes
5. sair de casa
6. esperar o ônibus às 08:20

Algoritmos

O que é um algoritmo?



Algoritmos

O que é um algoritmo?

```
def list_items(list):  
    for index, item in enumerate(list):  
        print('the index of {} is {}'.format(item, index))  
  
my_list = ['python', 'flask', 'music', 'phone', 'notebook', 'coffee']  
  
list_items(my_list)
```

Do ponto de vista da computação, é uma sequência de instruções que pode ser executada por uma máquina.

Para que a máquina consiga entender o que queremos que ela execute é necessário falarmos a “língua” dela.

Estruturas Lógicas

If (se)

O If é uma estrutura de condição que permite avaliar uma expressão e, de acordo com seu resultado, executar uma determinada ação.

```
1 | 1 idade = 18
2 | 2 if idade < 20:
3 | 3     print('Você é jovem!')
```

Estruturas Lógicas

If-else (se-senão)

Vimos anteriormente como utilizar o if para executar uma ação caso uma condição seja atendida. No entanto, nenhum comportamento específico foi definido para o caso de a condição não ser satisfeita. Quando isso é necessário, precisamos utilizar a reservada else.

```
1 | 1 idade = 18
2 | 2 if idade >= 18:
3 | 3     print('maior de idade')
4 | 4 else:
5 | 5     print('menor de idade')
```

Estruturas Lógicas

If-elif-else (se-ou se-senão)

Adicionalmente, se existir mais de uma condição alternativa que precisa ser verificada, devemos utilizar o elif para avaliar as expressões intermediárias antes de usar o else, da seguinte forma:

```
1 1 idade = 18
2 2 if idade < 12:
3 3     print('criança')
4 4 elif idade < 18:
5 5     print('adolescente')
6 6 elif idade < 60:
7 7     print('adulto')
8 8 else:
9 9     print('idoso')
```

Estruturas Lógicas

For (para cada)

Em algumas situações, é comum que uma mesma instrução (ou conjunto delas) precise ser executada várias vezes seguidas. Nesses casos, normalmente utilizamos um loop (ou laço de repetição), que permite executar um bloco de código repetidas vezes, enquanto uma dada condição é atendida.

```
1 1 nomes = ['Pedro', 'João', 'Leticia']
2 2 for n in nomes:
3 3     print(n)
4 4 >>>
5 5 Pedro
6 6 João
7 7 Leticia
```

While (Enquanto)

O comando `while`, por sua vez, faz com que um conjunto de instruções seja executado enquanto uma condição for atendida. Quando o resultado passa a ser falso, a execução é interrompida, saindo do loop, e passa para o próximo bloco.

```
1 | 01 contador = 0
2 | 02 while contador < 5:
3 | 03     print(contador)
4 | 04     contador = contador + 1
```

Estruturas Lógicas

Você sabe fazer omelete?

Qual seria o algoritmo?

Lembre-se de separar os ingredientes antes de começar a produção.



Direito, programação e culinária



Atividades

Dia 03 – Atividades



Breaking the Law

Copyright 2019 Roberto Novaes (www.robertonovaes.com.br)

Copyright 2019 SiLEX Sistemas (www.silexistemas.com.br)

Atribuição-NãoComercial-CompartilhaIgual 4.0 Internacional



Realização:



U F *m* G

Apoio:

MARCELO CANDIOTTO **MCSA**
Sociedade de Advogados



ASSOCIAÇÃO
BRASILEIRA DE
LAWTECHS &
LEGALTECHS