

FUNÇÕES

início

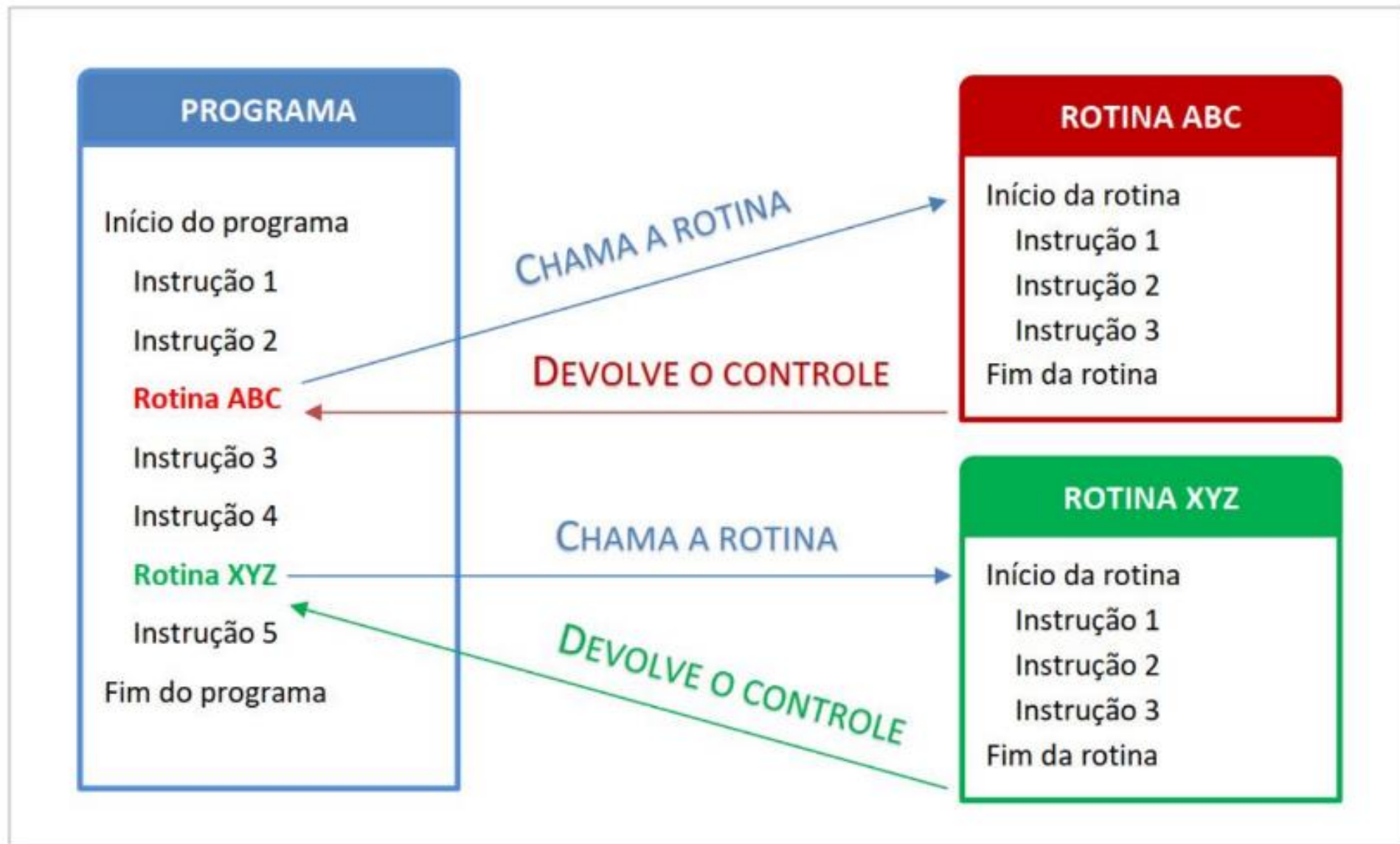
Introdução

- É mais fácil resolver um único problema complexo ou vários problemas simples?
- Imagine um programa como um editor de texto, que pode ter mais de 10 milhões de linhas de código, será que apenas uma pessoa construiu tudo sozinha?
- É útil diluir um problema grande em problemas menores e propor soluções para os problemas menores. Ao final as soluções são integradas e resolve-se o problema original!
- As funções (um tipo de rotina) são usadas para isso e ainda há outras vantagens.



O que é uma função/rotina?

- É uma sequência de instruções que executa alguma tarefa bem definida e que damos um nome;
- É comum que um programa contenha várias funções/rotinas, que são independentes do programa principal;
- Como são independentes, podem inclusive ser usadas em outros programas, poupando o retrabalho de escrever os mesmos códigos;
- Possibilitam códigos geralmente mais legíveis, organizados e, conseqüentemente, mais simples de serem mantidos.



Como escrever em Python

#Criação da função

```
def nome_da_funcao(parâmetro1, parâmetro2):  
    criação de variáveis  
    demais instruções  
    return
```

Detalhes importantes

- Existem dois conceitos fundamentais quando usamos funções: o momento em que a função é **criada** e o momento em que ela é **chamada**;
- Toda função deve ter sido criada (definida) antes de ser chamada (usada);
- Geralmente as funções são escritas no início do código-fonte;
- Existem funções que são disponibilizadas junto com o ambiente de programação e outras são criadas pelo próprio programador ou importadas de arquivos criados por outras pessoas.

```
# Função que calcula e exibe a média aritmética de três números.  
def exibe_media(n1, n2, n3):  
    soma = n1 + n2 + n3  
    print(soma / 3)
```

```
# Função que calcula e retorna como resposta a média aritmética  
# de três números.  
def retorna_media(num1, num2, num3):  
    soma = num1 + num2 + num3  
    return soma / 3
```

Detalhes importantes

- Toda função têm um nome, as regras são as mesmas que usamos para os identificadores de variáveis;
- Existem funções com ou sem parâmetros;
- Existem funções com ou sem retorno de valor*;
- Toda função, ao encerrar normalmente, devolve o controle (e potencialmente o valor de resposta) para a instrução em que foi chamada;
- Pense nas funções que foram projetadas para ter valor de retorno como **PERGUNTAS** e as funções que foram projetadas para não ter valor de retorno como **ORDENS**.

*Em Python há uma particularidade.