



Algoritmos e Programação de Computadores

Aula 06 - Funções

Prof. Dr. Rodolfo Carneiro Cavalcante
rodolfo.cavalcante@arapiraca.ufal.br
Universidade Federal de Alagoas
Campus de Arapiraca

Introdução



- Até agora temos realizado algumas chamadas de função
 - `print(argumentos)`
 - `len(lista)`
 - `lista.append(argumentos)`
 - `int(3,99)`
 - `float('3,14159')`

Introdução



- Funções são blocos de código que podem ser invocados de qualquer parte do código
- Operações já implementadas
- Evita necessidade de reimplementação
 - reuso de código
- Ex: `max(lista)`
- Boa prática de programação
 - facilita manutenibilidade de código

Introdução



- Funções são blocos de código que podem ser invocados de qualquer parte do código
- Operações já implementadas
- Evita necessidade de reimplementação
 - reuso de código
 - implemente uma vez, use sempre
- Boa prática de programação
 - facilita manutenibilidade de código

Introdução



Sistema de notas

Aprovado ou Reprovado

Funções



Função é composta por

- nome
- parâmetros (opcional)
- bloco de instruções
- retorno (opcional)

Funções



Sintaxe

```
def nome_funcao(parametros):  
    instrucao 1  
    ...  
    instrucao n  
    return value
```

- def: início da definição da função
- parâmetros: comunicação com a função
- indentação para demarcar bloco

Funções



Função mensagem de boas vindas

```
def boas_vindas(nome_usuario):  
    print('Ola, %s. Seja bem vindo ao sistema' %  
nome_usuario)  
  
boas_vindas('Rodolfo')
```


Funções



Calculadora

- Soma
- Subtração
- Multiplicação
- Divisão

Funções



Soma

```
def soma(a, b):  
    return a + b
```

Subtração

```
def subtracao(a, b):  
    return a - b
```

Funções



Multiplicação

```
def multiplicacao(a, b):  
    return a * b
```

Divisão

```
def divisao(a, b):  
    return a / b
```

Funções



Chamando as funções no programa principal

```
from calculadora import *  
  
numero1 = 3  
numero2 = 5  
sum = soma(numero1, numero2)  
print(sum)  
print(subtracao(numero1, numero2))  
print(multiplicacao(numero1, sum))  
print(divisao(sum, multiplicacao(numero1, sum)))
```

Funções



Divisão

- problema quando b é igual a 0

Funções



Divisão

- problema quando b é igual a 0

```
def divisao(a, b):  
    if b == 0:  
        return None  
    else:  
        return a / b
```

Funções



Valor de retorno

- Uma função pode retornar um valor após sua execução
- Esse valor de retorno será enviado para o ponto onde a função foi invocada
- Pode ser utilizado pelo código que invoca a função
- Basta utilizar o comando `return valor`

Funções



Vantagens do uso de funções

- Agrupar instruções
- Lógica das tarefas centralizadas em um único lugar
- Evitar reescrita de código
- Facilitar manutenção de código

Funções



Ex: Cálculo de imposto de renda

- Todo mês uma empresa precisa calcular quanto de imposto de renda de seus funcionários irá repassar para a receita federal
- Regras
 - Se o funcionário ganha até R\$ 1500,00 ele está isento de pagar imposto
 - Se ganha acima disso, precisa pagar 27% de imposto sobre o salário

Funções



Ex: Cálculo de imposto de renda

- Crie uma lista de salários de funcionários
- Calcule o imposto de renda para cada funcionário na lista

Funções



Ex: Cálculo de imposto de renda

```
def calcularImposto(salario):  
    if salario <= 1500:  
        imposto = 0  
    else:  
        imposto = 0.27*salario  
    return imposto  
  
lista_salarios = [1200, 1350, 1800, 1200, 2500,  
3400, 4500, 2300]  
for salario in salarios:  
    print(calcularImposto(salario))
```

Funções



Ex: Cálculo de imposto de renda

- Se a lógica de negócio mudar, basta alterar a função de cálculo do imposto
- Digamos que a isenção agora é somente para quem ganha até R\$ 1000,00
- E a alíquota do imposto passou a ser 32%
- Faça a alteração no código

Funções



Ex: Cálculo de imposto de renda

```
def calcularImposto(salario):  
    if salario <= 1500:  
        imposto = 0  
    else:  
        imposto = 0.32*salario  
    return imposto  
  
lista_salarios = [1200, 1350, 1800, 1200, 2500,  
3400, 4500, 2300]  
for salario in salarios:  
    print(calcularImposto(salario))
```

Funções



Ex: Cálculo do salário de funcionários

- Uma empresa precisa calcular o pagamento de seus funcionários
- Todo mês são contadas as horas trabalhadas por cada funcionário
- Cada funcionário tem um valor a ser pago por hora trabalhada
- Se a quantidade de horas trabalhadas excede 40 horas então é preciso calcular hora extra
- O valor da hora extra é 1.5 vezes a hora comum de trabalho

Funções



```
def calcularSalario(horas,valor_hora):  
    salario = 0  
    if horas <= 40:  
        salario = horas*valor_hora  
    else:  
        salario = 40*valor_hora +  
(horas-40)*(1.5*valor_hora)  
  
horas_trabalhadas = [45,32,28,50,55,60,40,30]  
valor_hora = 40  
for hora in horas_trabalhadas:  
    print(calcularSalario(hora,valor_hora))
```

Exercícios



1. Escreva um programa que possui uma função “maior”, que recebe uma lista e devolve o maior elemento na lista
2. Escreva um programa que possui uma função que recebe uma lista e um valor e verifica se existe o valor na lista
3. Escreva um programa que possui uma função que recebe uma lista e diz qual a soma máxima entre dois elementos da lista
4. Faça um algoritmo que copie o conteúdo de uma lista para outra, eliminando valores repetidos. Implemente funções para isso

Exercícios



5. Um professor teve uma ideia de como avaliar seus alunos em uma atividade que vale entre 0 e 10 de modo a incentivar a competição entre os alunos. Quem tirar a maior nota terá 10. Quem tirar a menor nota, terá 0. As outras notas serão algo entre 0 e 10 da seguinte forma:

$$\text{Nota} = ((\text{nota-min})/(\text{max} - \text{min})) * 10$$

Faça um programa para calcular as notas dos alunos segundo essa regra, utilizando funções

Exercícios



6. Uma palavra A é dita uma permutação de uma palavra B se os caracteres de A formam uma permutação dos caracteres de B

Ex: amor é uma permutação de roma

Ex: metro é uma permutação de morte

Faça uma função permutação que recebe duas palavras e verifica se uma é permutação da outra

Exercícios



7. Faça um programa que tenha duas funções, uma para calcular a média e outra para calcular a variância de um conjunto de números.
8. Faça um programa que recebe duas palavras A e B e verifica se B é uma substring de A.

Ex:

A	B	resposta
acabou	cabo	sim
abobora	bora	sim
calor	calo	sim
avisar	vida	não



Algoritmos e Programação de Computadores

Aula 06 - Funções

Prof. Dr. Rodolfo Carneiro Cavalcante
rodolfo.cavalcante@arapiraca.ufal.br