

Avaliação Prática 3

Valor total: **10 pontos**

João Guilherme Maia de Menezes
Jefersson Alex dos Santos

23/11/2021

Regras de Conduta

- Esta é uma atividade avaliativa que deve ser realizada de maneira individual e sem consulta.
- Celulares devem permanecer desligados durante a atividade.
- Espera-se que o aluno realize suas atividades com honestidade e integridade.
- Falhas de conduta, como cópia de atividades de colegas, serão punidas com dedução parcial ou total da nota, além de outras penalidades, segundo as normas do Colegiado de Graduação.

Instruções para Submissão

Na avaliação prática de hoje, você terá que elaborar 5 programas para resolver problemas diversos, conforme descrito abaixo. Cada uma das soluções deverá ser implementada em seu próprio arquivo com extensão `.py`. Por exemplo, a solução para o problema 1 deverá ser implementada em um arquivo chamado `problema1.py`, a solução para o problema 2 deverá ser implementada no arquivo `problema2.py` e assim por diante. Finalmente, submeta cada um dos arquivos pelo Moodle.

Dica: se você tiver problemas com caracteres especiais (caracteres com acentos, por exemplo), adicione a linha abaixo na primeira linha de todos os arquivos `*.py`

```
# -*- coding: utf-8 -*-
```

Problema 1 - (2 pontos)

Escreva um função chamada `fizz_buzz` que recebe como **parâmetro** um número inteiro e **retorna** um número inteiro ou uma string de acordo com as regras abaixo:

1. Se o número é divisível por 3, a função deve retornar a string "Fizz".
2. Se o número é divisível por 5, a função deve retornar a string "Buzz".
3. Se o número é divisível por 3 e por 5, a função deve retornar a string "FizzBuzz".
4. Senão, a função deve retornar o mesmo número inteiro que recebeu como parâmetro.

Observação 1: O nome da função deve ser exatamente como especificado acima e seu arquivo deve obrigatoriamente se chamar `problema1.py`.

Observação 2: Seu programa deve conter apenas a função descrita acima. Você não precisa realizar a entrada e saída de dados (não precisa usar as funções `input()` e `print()`).

Exemplo 1 de execução do programa:

Digite um número: 10

Buzz

Exemplo 2 de execução do programa:

Digite um número: 15

FizzBuzz

Exemplo 3 de execução do programa:

Digite um número: 2

2

Problema 2 - (2 pontos)

Escreva uma função chamada `consumo` que recebe como **parâmetros** a distância percorrida em quilômetros e a quantidade de litros de gasolina consumidos por um veículo para percorrer tal distância. Sua função deve **retornar** uma mensagem de acordo com a tabela abaixo:

Consumo	Km/l	Mensagem
Menor que	8	Venda o carro!
Entre	8 e 12	Econômico!
Maior que	12	Super econômico!

Observação 1: O nome da função deve ser exatamente como especificado acima e seu arquivo deve obrigatoriamente se chamar `problema2.py`.

Observação 2: Seu programa deve conter apenas a função descrita acima. Você não precisa realizar a entrada e saída de dados (não precisa usar as funções `input()` e `print()`).

Exemplo 1 de execução do programa:

Digite a distância: **100**

Digite a quantidade de gasolina consumida: **10**

Econômico!

Exemplo 2 de execução do programa:

Digite a distância: **30**

Digite a quantidade de gasolina consumida: **4.5**

Venda o carro!

Problema 3 - (2 pontos)

Escreva uma função chamada `estacionamento` que recebe como **parâmetros** a hora e minuto de entrada e hora e minuto de saída de um estacionamento e **retorna** o valor total devido de acordo com as seguintes regras:

Quantidade de Horas	Tarifa
Até duas horas	R\$ 1.00 para cada hora
Entre três e quatro horas	R\$ 1.40 para cada hora
Acima de quatro horas	R\$ 2.00 para cada hora

O número de horas a pagar é sempre inteiro e arredondado por excesso. Deste modo, quem estacionar durante 61 minutos pagará por 2 horas. Os momentos de chegada e partida do estacionamento são apresentados na forma de pares de inteiros, representando horas e minutos. Por exemplo, o par 12 50 representará "dez para a uma da tarde". Admite-se que a chegada e a partida se dão com intervalo não superior a 24 horas.

Observação 1: O nome da função deve ser exatamente como especificado acima e seu arquivo deve obrigatoriamente se chamar `problema3.py`.

Observação 2: Seu programa deve conter apenas a função descrita acima. Você não precisa realizar a entrada e saída de dados (não precisa usar as funções `input()` e `print()`).

Exemplo 1 de execução do programa:

Digite a hora de chegada: **18**

Digite o minuto de chegada: **50**

Digite a hora da partida: **22**

Digite o minuto da partida: **49**

Preço: R\$ **5.60**

Exemplo 2 de execução do programa:

Digite a hora de chegada: **20**

Digite o minuto de chegada: **30**

Digite a hora da partida: **8**

Digite o minuto da partida: **00**

Preço: R\$ **24.00**

Problema 4 - (2 pontos)

Elabore uma função chamada `media` que receba como parâmetros três notas de um aluno, e uma letra. Se a letra for "A", a função deverá retornar a média aritmética das notas do aluno; se for "P", deverá retornar a média ponderada, com pesos 5, 3 e 2 para cada nota, respectivamente.

Observação 1: O nome da função deve ser exatamente como especificado acima e seu arquivo deve obrigatoriamente se chamar `problema4.py`.

Observação 2: Seu programa deve conter apenas a função descrita acima. Você não precisa realizar a entrada e saída de dados (não precisa usar as funções `input()` e `print()`).

Exemplo 1 de execução do programa:

Digite a primeira nota: **1.0**
Digite a segunda nota: **2.0**
Digite a terceira nota: **3.0**
Digite o tipo da média: **A**
Média: **2.00**

Exemplo 2 de execução do programa:

Digite a primeira nota: **1.0**
Digite a segunda nota: **2.0**
Digite a terceira nota: **3.0**
Digite o tipo da média: **P**
Média: **1.70**

Problema 5 - (2 pontos)

Faça uma função chamada `operacao` que receba como parâmetros dois valores numéricos e um símbolo. Este símbolo representará uma operação que deve ser efetuada com os números. Se o símbolo for "+", a função deve retornar a adição; se for "-", a subtração; se for "*", a multiplicação; e se for "/", a divisão.

Observação 1: O nome da função deve ser exatamente como especificado acima e seu arquivo deve obrigatoriamente se chamar `problema5.py`.

Observação 2: Seu programa deve conter apenas a função descrita acima. Você não precisa realizar a entrada e saída de dados (não precisa usar as funções `input()` e `print()`).

Exemplo 1 de execução do programa:

Digite o primeiro valor: **1.0**
Digite o segundo valor: **2.0**
Digite a operação: **+**
Resultado: **3.0**

Exemplo 2 de execução do programa:

Digite o primeiro valor: **1.0**
Digite o segundo valor: **2.0**
Digite a operação: -
Resultado: **-1.00**

Exemplo 3 de execução do programa:

Digite o primeiro valor: **2.0**
Digite o segundo valor: **2.0**
Digite a operação: *
Resultado: **4.00**

Exemplo 4 de execução do programa:

Digite o primeiro valor: **10.0**
Digite o segundo valor: **2.0**
Digite a operação: /
Resultado: **5.00**