



CENTRO UNIVERSITÁRIO INTERNACIONAL UNINTER
ESCOLA SUPERIOR POLITÉCNICA

BACHARELADO EM ENGENHARIA DE SOFTWARE
LÓGICA DE PROGRAMAÇÃO E ALGORITMOS

ATIVIDADE PRÁTICA

NOME: FILIPPE CARLOS V. DE FREITAS – RU: 3749385
NOME PROFESSOR: VINICIOS BOZZOBON BORIN

BARUERI - SÃO PAULO
27 DE JULHO DE 2021

Exercício 1:

Escreva um programa que leia o nome de um aluno e sua nota final. Em seguida, informe o conceito conforme a tabela abaixo. A saída do programa deve exibir na tela uma frase com o padrão descrito a seguir:

Nome do aluno: Fábio José

Nota final: 73.4

Frase a ser exibida: O aluno Fabio José tirou nota 3.5 e se enquadra no conceito D

Nota	Conceito
De 0,0 a 2,9	E
De 3,0 a 4,9	D
De 5,0 a 6,9	C
De 7 a 8,9	B
De 9,0 a 10	A

Todos os dados devem ser lidos do teclado, sendo que o nome do aluno é uma string e a nota final é um número real. Não é necessário armazenar os dados em uma estrutura de dados, basta imprimir na tela.

Coloque todo o seu programa dentro de um laço de repetição e faça-o se encerrar quando uma determinada condição for satisfeita. A condição fica a seu critério, como por exemplo, encerrar o programa ao digitar a palavra sair, ou então uma nota inválida.

Imprima na tela um teste do seu programa utilizando o seu nome e os dois últimos dígitos do seu RU para a nota.



exercicio 1.py

```

21 #Menu de seleção para saber se o usuario quer iniciar o programa ou não
22 começar = int(input('Deseja inserir dados? 0 - Não    1 - Sim '))
23
24 # Laço de repetição para verificar se inicia ou encerra
25 while começar == 1:
26     aluno = input("Digite nome do aluno: ")
27     nota = float(input("Digite a nota final do aluno de 0 - 10: "))
28
29     # Inicio da verificação de nota e atribuição do conceito
30     if 0 <= nota < 3:
31         conceito = 'E'
32     elif 3 <= nota < 5:
33         conceito = 'D'
34     elif 5 <= nota < 7:
35         conceito = 'C'
36     elif 7 <= nota < 9:
37         conceito = 'B'
38     elif 9 <= nota <= 10:
39         conceito = 'A'
40     else:# Essa 1ª condição se a nota for maior que 10 - Minha condição para que o programa se encerre
41         print('Nota inválida, encerrando o programa.')
42         exit() # Sair do laço de repetição
43     # Resultado final de acordo com a nota do aluno e seu conceito
44     print(f"O aluno {aluno} tirou a nota {nota} e se enquadra no conceito {conceito}")
45     começar = int(input('Inserir dados? 0 - Não    1 - Sim '))
46 else: # Essa 2ª condição só é valida se no inicio o usuario - Minha condição para que o programa se encerre
47     print ("Obrigado por usar nosso programa")
48
49
50

```

Exercício 1

```

exercicio 1.py
exercicio 1.py > ...
37     conceito = 'B'
38     elif 9 <= nota <= 10:
39         conceito = 'A'
40     else: # Essa 1ª condição se a nota for maior que 10 - Minha condição para que o programa se encerre
41         print('Nota inválida, encerrando o programa.')
42         exit() # Sair do laço de repetição
43     # Resultado final de acordo com a nota do aluno e seu conceito
44     print(f"O aluno {aluno} tirou a nota {nota} e se enquadra no conceito {conceito}")
45     comecar = int(input('Inserir dados? 0 - Não      1 - Sim '))
46 else: # Essa 2ª condição só é valida se no inicio o usuario - Minha condição para que o programa se encerre
47     print ("Obrigado por usar nosso programa")
48
49
50
51

```

PROBLEMS
OUTPUT
TERMINAL
DEBUG CONSOLE

Python + ^ ^ X

Windows PowerShell

Copyright (C) Microsoft Corporation. Todos os direitos reservados.

Experimente a nova plataforma cruzada PowerShell <https://aka.ms/pscore6>

```

PS C:\Users\Filippe Freitas\Desktop\Exercicios Py> & "C:/Users/Filippe Freitas/AppData/Local/Programs/Python/Python39/python.exe" "c:/Users/Filippe Freitas/Desktop/Exercicios Py/exercicio 1.py"
Deseja inserir dados? 0 - Não      1 - Sim 0
Obrigado por usar nosso programa
PS C:\Users\Filippe Freitas\Desktop\Exercicios Py> & "C:/Users/Filippe Freitas/AppData/Local/Programs/Python/Python39/python.exe" "c:/Users/Filippe Freitas/Desktop/Exercicios Py/exercicio 1.py"
Deseja inserir dados? 0 - Não      1 - Sim 1
Digite nome do aluno: Felipe Carlos
Digite a nota final do aluno de 0 - 10: 9.5
O aluno Felipe Carlos tirou a nota 9.5 e se enquadra no conceito A
Inserir dados? 0 - Não      1 - Sim 1
Digite nome do aluno: Felipe Carlos
Digite a nota final do aluno de 0 - 10: 20
Nota inválida, encerrando o programa.
PS C:\Users\Filippe Freitas\Desktop\Exercicios Py>

```

Exercício 1

Exercício 2:

Faça uma **função** que receba o nome e sobrenome de uma pessoa e retorne a primeira letra de seu nome e seu sobrenome concatenando com a string @algoritmos.com.br. No algoritmo principal deverá ser apresentada a mensagem ao usuário contendo seu nome completo e seu email.

Imprima na tela um teste do seu programa utilizando o seu nome e sobrenome concatenado com os dois últimos dígitos do seu RU.

Sra Luciane Kanashiro, seu email é ikanashiro16@algoritmos.com.br

```

exercicio2.py
exercicio2.py > geracao_email

9 # meu RU 749385
10 # Função para gerar e-mail com strings formatadas
11 def geracao_email(nome:str, sobrenome:str, ru: str):
12     # Validação por if para não aceitar strings vazias
13     if len(nome) > 0 and len(sobrenome) > 0:
14         return 'Sr. ' + nome + ' ' + sobrenome + ', seu e-mail é ' + nome[0].lower()# Tive que pular uma linha se não não ia caber
15         + sobrenome.lower() + ru + '@algoritmos.com.br' # Lower serve para alterar o tamanho da primeira letra para minuscúlo
16     else:
17         return None
18
19 #Input para receber o nome e sobrenome do usuário:
20 primeiro_nome = input('Digite o primeiro nome:')
21 ultimo_sobrenome = input('Digite o último sobrenome:')
22 ru = "85" # Meu RU
23 print(1 * '\n') #Quebra de linha para melhor visualização dos resultados
24 print(geracao_email(primeiro_nome, ultimo_sobrenome, ru))
25 print(1 * '\n') #Quebra de linha para melhor visualização dos resultados
26

```

PROBLEMS
OUTPUT
TERMINAL
DEBUG CONSOLE

Python

+

^

x

Windows PowerShell

Copyright (C) Microsoft Corporation. Todos os direitos reservados.

Experimente a nova plataforma cruzada PowerShell <https://aka.ms/pscore6>

PS C:\Users\Filippe Freitas\Desktop\Exercicios Py> & "C:/Users/Filippe Freitas/AppData/Local/Programs/Python/Python39/python.exe" "c:/Users/Filippe Freitas/Desktop/Exercicios Py/exercicio2.py"

Digite o primeiro nome:Filippe

Digite o último sobrenome:Carlos

Sr. Felipe Carlos, seu e-mail é fcarlos85@algoritmos.com.br

PS C:\Users\Filippe Freitas\Desktop\Exercicios Py>

Exercício 2

Exercício 3:

Um canal de jogos do youtube está fazendo um sorteio para angariar doações para pessoas em situação de vulnerabilidade social. A cada 10,00 doado o nome da pessoa é inserido em uma lista de sorteio, por exemplo:

Ruth doou 20,00;

Maria doou 30,00;

Fernando doou 50,00;

A lista de sorteio estará com os valores:

```
listaSorteio = ['Ruth', 'Ruth', 'Maria', 'Maria', 'Maria', 'Fernando', 'Fernando',  
'Fernando', 'Fernando', 'Fernando']
```

Implemente um programa para cadastrar o nome das pessoas que doaram. O programa deve embaralhar a lista, sortear o ganhador e imprimir o seu nome.

Imprima na tela um teste do seu programa utilizando como primeiro doador o seu nome e os dois últimos dígitos do seu RU para o valor doado. Não se esqueça de imprimir também a lista de sorteio.

Exercício 3

exercicio3.py •

exercicio3.py > ...

```
25
26 import random # Importação da biblioteca
27 #Variavel doadores inicia com uma lista vazia para ser gravada de acordo com a sua doação
28 colaboradores = []
29 #Função de cadastro que recebe o valor da doação em float e nome str
30 def cadastra(nome:str, doacao:float):
31     colaboradores.extend( ((nome + ' ') * int(doacao//10)).split() )
32     return
33 #função que embaralha a lista de doadores retornando um dos valores
34 def sorteio():
35     random.shuffle(colaboradores)
36     print(f'Lista embaralhada: {colaboradores}')
37     print(1 * '\n') # Quebra de linha para melhor visualização dos resultados
38     return random.choice(colaboradores)
39 # Valdador para saber se ainda quer continuar com o processo
40 opcao = int(input('Cadastrar doador? 0 - Não    1 - Sim '))
41 while opcao == 1:
42     doador = input('Insira o nome do doador: ')
43     valor = float(input('Insira o valor da doação: '))
44     cadastra(doador, valor)
45     # Valdador para saber se ainda quer continuar com o processo
46     opcao = int(input('Cadastrar doador? 0 - Não    1 - Sim '))
47
48 # Dados aceitos a função cadastrar_doador se torna True incluindo novos doadores na lista mantendo os antigos
49 if len(colaboradores) > 0:
50     print(f'Lista de doadores para sorteio: {colaboradores}')
51     print(1 * '\n') #Quebra de linha para melhor visualização dos resultados
52     print(f'O vencedor(a) foi: {sorteio()}')
53     print(1 * '\n') #Quebra de linha para melhor visualização dos resultados
54
```


exercício3.py X

exercício2.py

exercício3.py > cadastra

```
42 doador = input('Insira o nome do doador: ')
43 valor = float(input('Insira o valor da doação: '))
44 cadastra(doador, valor)
45 # Validador para saber se ainda quer continuar com o processo
46 opcao = int(input('Cadastrar doador? 0 - Não      1 - Sim '))
47
48 # Dados aceitos a função cadastrar_doador se torna True incluindo novos doadores na lista mantendo os antigos
49 if len(colaboradores) > 0:
50     print(f'Lista de doadores para sorteio: {colaboradores}')
51     print(1 * '\n') #Quebra de linha para melhor visualização dos resultados
52     print(f'O vencedor(a) foi: {sorteio()}')
53     print(1 * '\n') #Quebra de linha para melhor visualização dos resultados
54
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

Python + ^ X

Experimente a nova plataforma cruzada PowerShell <https://aka.ms/pscore6>

```
PS C:\Users\Filippe Freitas\Desktop\Exercicios Py> & "C:/Users/Filippe Freitas/AppData/Local/Programs/Python/Python39/python.exe" "c:/Users/Filippe Freitas/Desktop/Exercicios Py/exercicio3.py"
Cadastrar doador? 0 - Não      1 - Sim 0
PS C:\Users\Filippe Freitas\Desktop\Exercicios Py> & "C:/Users/Filippe Freitas/AppData/Local/Programs/Python/Python39/python.exe" "c:/Users/Filippe Freitas/Desktop/Exercicios Py/exercicio3.py"
Cadastrar doador? 0 - Não      1 - Sim 1
Insira o nome do doador: Felipe
Insira o valor da doação: 85
Cadastrar doador? 0 - Não      1 - Sim 1
Insira o nome do doador: Maria
Insira o valor da doação: 10
Cadastrar doador? 0 - Não      1 - Sim 0
Lista de doadores para sorteio: ['Filippe', 'Filippe', 'Filippe', 'Filippe', 'Filippe', 'Filippe', 'Filippe', 'Filippe', 'Filippe', 'Maria']
```

```
Lista embaralhada: ['Filippe', 'Filippe', 'Maria', 'Filippe', 'Filippe', 'Filippe', 'Filippe', 'Filippe', 'Filippe', 'Filippe']
```

```
O vencedor(a) foi: Maria
```

```
PS C:\Users\Filippe Freitas\Desktop\Exercicios Py> |
```

Exercício 3

Exercício 4:

Considere a tabela a seguir referente a produtos armazenados em um depósito, em que são considerados o estoque atual de cada produto e o estoque mínimo necessário.

Código	Estoque	Mínimo
1	35	20
5	75	50
2	43	45
3	26	18
20	35	20

Armazene as informações acima em uma estrutura de lista com dicionário, substituindo a primeira linha com os dados: no campo código coloque o primeiro dígito do seu RU, no estoque os dois dígitos seguintes de seu RU, e no campo mínimo os dois últimos dígitos do seu RU.

Por exemplo, tendo o RU: **1**23**4**56

Código	Estoque	Mínimo
1	23	56
5	75	50
2	43	45
3	26	18
20	35	20

exercicio4.py > ...

```

42
43 lista = [] # Lista começa vazia
44 dicionario = {}
45
46 # Função para cadastrar produto via append
47 def cadastra_produto(produto_para_cadastrar: dict):
48     lista.append(produto_para_cadastrar)
49     return
50
51 # Menu para saber se quer cadastrar
52 opcao = int(input('Cadastrar produto? 0 - Não    1 - Sim '))
53 while opcao == 1:
54     produto_novo = {} # Produto novo é uma variavel com dicionario vazio
55
56     produto_novo['codigo'] = int(input('Digite o código do produto: '))
57
58     #Validador se o cod não é zero
59     if produto_novo['codigo'] == 0:
60         print('Código 0, encerrando cadastro de produtos.')
61         break # Sai desse laço de repetição
62
63     # Inicia o input
64     produto_novo['estoque'] = int(input('Digite a quantidade em estoque: '))
65     produto_novo['minimo'] = int(input('Digite a quantidade mínima do estoque: '))
66
67     cadastra_produto(produto_novo)
68
69     #Validador se quer caastrar mais produto
70     opcao = int(input('Cadastrar produto? 0 - Não
71

```

Exercício 4

exercicio4.py > ...

```

68
69     #Validador se quer caastrar mais produto
70     opcao = int(input('Cadastrar produto? 0 - Não      1 - Sim '))
71
72 # Validador se a lista for >0 com retorno em tela com a tabela
73 if len(lista) > 0:
74     print('Lista de produtos por código em ordem crescente:')
75     print("Código".center(16), end='') # end faz com que se continue na msm linha
76     print("Estoque".center(16), end='') # end faz com que se continue na msm linha
77     print("Mínimo".center(16))
78
79 # PArá cada prod nessa lista já ordenada eu mostro na tela
80 for produto in sorted(lista, key=lambda item: item['codigo']):
81     print(str(produto['codigo']).center(16), end='') # end faz com que se continue na msm linha
82     print(str(produto['estoque']).center(16), end='') # end faz com que se continue na msm linha
83     print(str(produto['minimo']).center(16))
84 else:
85     print('Que pena, você ainda não cadastrou nenhum produto.')
86
87

```

Exercício 4

```
exercicio4.py • exercicio2.py
exercicio4.py > ...

75 # Validador se a lista for >0 com retorno em tela com a tabela
76 if len(lista) > 0:
77     print('Lista de produtos por código em ordem crescente:')
78     print("Código".center(16), end='') # end faz com que se continue na msm linha
79     print("Estoque".center(16), end='') # end faz com que se continue na msm linha
80     print("Mínimo".center(16))
81
82     # Para cada prod nessa lista ja ordenada eu mostro na tela
83     for produto in sorted(lista, key=lambda item: item['codigo']):
84         print(str(produto['codigo']).center(16), end='') # end faz com que se continue na msm linha
85         print(str(produto['estoque']).center(16), end='') # end faz com que se continue na msm linha
86         print(str(produto['minimo']).center(16))
87 else:
88     print('Que pena, você ainda não cadastrou nenhum produto.')
89
90
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE Python + ^ x

y/exercicio4.py"
Cadastrar produto? 0 - Não 1 - Sim 0
Que pena, você ainda não cadastrou nenhum produto.
PS C:\Users\Filippe Freitas\Desktop\Exercicios Py> & "C:/Users/Filippe Freitas/AppData/Local/Programs/Python/Python39/python.exe" "c:/Users/Filippe Freitas/Desktop/Exercicios Py/exercicio4.py"
Cadastrar produto? 0 - Não 1 - Sim 1
Digite o código do produto: 7
Digite a quantidade em estoque: 49
Digite a quantidade mínima do estoque: 85
Cadastrar produto? 0 - Não 1 - Sim 1
Digite o código do produto: 10
Digite a quantidade em estoque: 20
Digite a quantidade mínima do estoque: 30
Cadastrar produto? 0 - Não 1 - Sim 0
Lista de produtos por código em ordem crescente:
Código Estoque Mínimo
7 49 85
10 20 30
PS C:\Users\Filippe Freitas\Desktop\Exercicios Py> |

Exercício 4