

**ANO  
2025**



**UNINTER**

**ATIVIDADE PRÁTICA**  
**MÓDULO C**  
**LÓGICA DE PROGRAMAÇÃO**  
**E ALGORITMOS**

**BRUNO ELIAKIM. RU:5558147**

**Prof. Me. Bruno Kostiuk**

# INSTRUÇÕES

**SOMENTE INFORMATIVO. PODE APAGAR AO ENTREGAR O TRABALHO.**

Esta atividade deve ser desenvolvida individualmente em linguagem Python, não sendo permitido a utilização de Inteligência Artificial para resolução das questões. A entrega deve ser feita na seção Trabalhos do Univirtus. Depois do arquivo enviado, não há possibilidade de reenvio. Verifique o arquivo enviado antes de confirmar a entrega. Só envie seu trabalho quando tiver certeza de que está tudo correto. Preste muita atenção ao prazo! Não haverá prorrogação.

Só serão aceitas submissões em formato doc ou docx. Pois estes formatos mantêm a indentação do Python, facilitando a execução dos códigos de vocês.

## **Em caso de dificuldade na elaboração ou publicação do trabalho, envie uma tutoria!**

Iremos auxiliar na tutoria, desde que não envolva fazer uma correção prévia do trabalho. Na tutoria, nós damos dicas de como solucionar o problema, mas a solução é com vocês alunos.

As quatro questões não poderão ser feitas por qualquer tipo de Inteligência Artificial. Caso seja identificado esse método de solução, o trabalho receberá nota zero automaticamente.

A proposta deste trabalho é que o aluno coloque em prática e demonstre suas competências e habilidades adquiridas ao longo da disciplina. Para isso, pede-se neste trabalho a confecção e entrega das quatro questões a seguir:

## QUESTÃO 1 de 4 – Conteúdos até Aula 3

**Enunciado:** Imagina-se que você é um dos programadores responsáveis pela construção de app para uma empresa X que vende Planos de Saúde. Uma das estratégias dessa empresa X é cobrar um valor diferente com base na idade do cliente, conforme a **listagem abaixo**:

- Se a idade for **maior ou igual** que **0** e **menor que 19**, o valor será de **100%** do **valor base** do plano (100 / 100);
- Se a idade for **maior ou igual** que **19** e **menor que 29**, o valor será de **150%** do **valor base** do plano (150 / 100);
- Se a idade for **maior ou igual** que **29** e **menor que 39**, o valor será de **225%** do **valor base** do plano (225 / 100);
- Se a idade for **maior ou igual** que **39** e **menor que 49**, o valor será de **240%** do **valor base** do plano (240 / 100);
- Se a idade for **maior ou igual** que **49** e **menor que 59**, o valor será de **350%** do **valor base** do plano (350 / 100);
- Se a idade for **maior ou igual** que **59**, o valor será de **600%** do **valor base do plano** (600 / 100);

O valor mensal do plano é calculado da seguinte maneira:

$$valorMensal = valorBase * porcentagem$$

**Exemplo:** Se o **valorBase** informado for **100.00** e a **idade** for **45** anos (**240%** segundo a tabela acima)

$$valorMensal = 100.00 * \left( \frac{240}{100} \right) = R \$ 240.00$$

**Elabore um programa em Python que:**


- Deve-se implementar o print com uma mensagem de boas-vindas que apareça o seu **nome e sobrenome** (somente print, não usar input aqui). [EXIGÊNCIA DE CÓDIGO 1 de 6];
- Deve-se implementar o input do **valorBase** do plano e da **idade** do cliente [EXIGÊNCIA DE CÓDIGO 2 de 6];
- Deve-se implementar as regras de valores **conforme a enunciado acima** (obs.: atente-se as condições de menor, igual e maior) [EXIGÊNCIA DE CÓDIGO 3 de 6];
- Deve-se implementar o **valorMensal** [EXIGÊNCIA DE CÓDIGO 4 de 6];
- Deve-se implementar as estruturas **if, elif e else (todas elas)** [EXIGÊNCIA DE CÓDIGO 5 de 6];
- Deve-se inserir comentários relevantes no código [EXIGÊNCIA DE CÓDIGO 6 de 6];

**Teste seu código atendendo as seguintes exigências:**

- Deve-se apresentar na saída de console uma mensagem de boas-vindas com seu **nome e sobrenome** [EXIGÊNCIA DE SAÍDA DE CONSOLE 1 de 2];

- H. Deve-se apresentar na saída de console a utilização do sistema informando uma **idade maior ou igual a 29 anos**, apresentando na saída de console o **valorMensal** do plano [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 2];

#### EXEMPLO DE SAÍDA DE CONSOLE:



```
Bem vindo ao Sistema do Bruno KostiuK
Informe o valor Base do plano: R$ 134.05
Informe a idade do cliente: 34
O valor mensal do plano é de: R$ 301.61
```

Figura 1: Exemplo de saída de console que o aluno deve fazer. Em que se perguntar o valorBase do plano (pode ser qualquer valor) e a idade (maior ou igual a 29 anos [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 2]), e é apresentado o valorMensal.

## ***Apresentação de Código da Questão 1:***

SUBSTITUIR ESSE TEXTO QUE ESTÁ EM VERMELHO PELO SEU CÓDIGO DO EXERCÍCIO 1.

NÃO ESQUECER DE CUMPRIR AS EXIGÊNCIAS DE CÓDIGO!!

O CÓDIGO DEVE ESTAR IDENTADO!! (SE NÃO ESTIVER, ZERA ESSA PARTE DA QUESTÃO!)

SERÃO ACEITOS SOMENTE CÓDIGOS NO FORMATO TEXTO (NADA DE IMAGEM NEM PRINT, ZERA ESSA PARTE DA QUESTÃO!).

## *Apresentação de Saída do Console da Questão 1:*

SUBSTITUIR ESSE TEXTO QUE ESTÁ EM LARANJA PELA A SAÍDA DO CONSOLE DO EXERCÍCIO 1  
NÃO ESQUECER DE CUMPRIR AS EXIGÊNCIAS  
SERÁ ACEITO SOMETE **SAÍDAS DO CONSOLE NO FORMATO IMAGEM** (NADA DE TEXTO AQUI! ZERA  
ESSA PARTE DA QUESTÃO!)

## QUESTÃO 2 de 4 - Conteúdo até aula 04

**Enunciado:** Você e sua equipe de programadores foram contratados para desenvolver um app de vendas para uma Pizzaria que vende sabores de Pizzas Doces e Pizzas Salgadas. Você ficou com a parte de desenvolver a interface do cliente para retirada do produto.

A Loja possui seguinte relação:

- Tamanho **P**: Pizza Salgada (**PS**) custa 30 reais e a Pizza Doce (**PD**) custa 34 reais;
- Tamanho **M**: Pizza Salgada (**PS**) custa 45 reais e a Pizza Doce (**PD**) custa 48 reais;
- Tamanho **G**: Pizza Salgada (**PS**) custa 60 reais e a Pizza Doce (**PD**) custa 66 reais;

**Elabore um programa em Python que:**

- Deve-se implementar o print com uma mensagem de boas-vindas que apareça o seu **nome e sobrenome** (somente print, não usar input aqui). Além do seu nome completo, deve-se implementar um **print com um Menu** para o cliente. [EXIGÊNCIA DE CÓDIGO 1 de 8];
- Deve-se implementar o input do **sabor** (PS/PD) e o print "Sabor inválido. Tente novamente" se o usuário entra com valor diferente de PS e PD [EXIGÊNCIA DE CÓDIGO 2 de 8];
- Deve-se implementar o input do **tamanho** (P/M/G) e o print "Tamanho inválido. Tente novamente" se o usuário com entra valor diferente de P, M ou G [EXIGÊNCIA DE CÓDIGO 3 de 8];
- Deve-se implementar **if, elif e/ou else**, utilizando o modelo **aninhado** (aula 3 – Tema 4) com cada uma das combinações de **sabor** e **tamanho** [EXIGÊNCIA DE CÓDIGO 4 de 8];
- Deve-se implementar um **acumulador** para somar os valores dos pedidos (valor total do pedido) [EXIGÊNCIA DE CÓDIGO 5 de 8];
- Deve-se implementar o input com a pergunta: "Deseja pedir mais alguma coisa?". Se sim **repetir a partir do item B**, senão encerrar o programa executar o print do **acumulador** [EXIGÊNCIA DE CÓDIGO 6 de 8];
- Deve-se implementar as estruturas de **while, break, continue (todas elas)** [EXIGÊNCIA DE CÓDIGO 7 de 8];
- Deve-se inserir comentários relevantes no código [EXIGÊNCIA DE CÓDIGO 8 de 8];

**Teste seu código atendendo as seguintes exigências:**

- Deve-se apresentar na saída de console uma mensagem de boas-vindas com o seu **nome e sobrenome** e o menu para o cliente conhecer as opções [EXIGÊNCIA DE SAÍDA DE CONSOLE 1 de 4];
- Deve-se apresentar na saída de console um pedido em que o usuário errou o **sabor** [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 4];
- Deve-se apresentar na saída de console um pedido em que o usuário errou o **tamanho** [EXIGÊNCIA DE SAÍDA DE CONSOLE 3 de 4];
- Deve-se apresentar na saída de console um pedido com **duas opções sabores diferentes e com tamanhos diferentes** [EXIGÊNCIA DE SAÍDA DE CONSOLE 4 de 4];

## EXEMPLO DE SAÍDA DE CONSOLE:

```
➡ ----- Bem-vindo a Pizzaria do Bruno Kostiuk -----  
-----Cardápio-----  
-----  
---| Tamanho | Pizza Salgada(PS) | Pizza Doce(PD) | ---  
---| P        | R$ 30.00          | R$ 34.00        | ---  
---| M        | R$ 45.00          | R$ 48.00        | ---  
---| G        | R$ 60.00          | R$ 66.00        | ---  
-----  
Entre com o sabor desejado (PS/PD): PG  
Sabor inválido. Tente novamente  
  
Entre com o sabor desejado (PS/PD): PS  
Entre com o tamanho desejado (P/M/G): EXGG  
Tamanho inválido. Tente novamente  
  
Entre com o sabor desejado (PS/PD): PS  
Entre com o tamanho desejado (P/M/G): G  
Você pediu uma Pizza Salgada no tamanho G: R$ 60.00  
  
Deseja mais alguma coisa? (S/N): S  
Entre com o sabor desejado (PS/PD): PD  
Entre com o tamanho desejado (P/M/G): M  
Você pediu um Filé de Frango no tamanho M: R$ 48.00  
  
Deseja mais alguma coisa? (S/N): N  
  
O valor total a ser pago: R$ 108.00
```

Mensagem com seu nome completo e Menu de opções (cardápio)

Usuário errou o sabor

Usuário errou o tamanho

Pedido com 2 itens de tamanho e sabores diferentes

**Figura 2: Exemplo de saída de console que o aluno deve fazer. Em que se perguntar o sabor ao tamanho. Há uma tentativa de pedido que se erro o sabor e outra que se errou o tamanho. Há também dois pedidos com sabores e tamanhos diferentes.**



## *Apresentação de **Código da Questão 2:***

SUBSTITUIR ESSE TEXTO QUE ESTÁ EM VERMELHO PELO SEU CÓDIGO DO EXERCÍCIO 2.

NÃO ESQUECER DE CUMPRIR AS EXIGÊNCIAS DE CÓDIGO!!

O CÓDIGO DEVE ESTAR IDENTADO!! (SE NÃO ESTIVER, ZERA ESSA PARTE DA QUESTÃO!)

SERÃO ACEITOS SOMENTE CÓDIGOS NO FORMATO TEXTO (NADA DE IMAGEM NEM PRINT, ZERA ESSA PARTE DA QUESTÃO!).

## *Apresentação de Saída do Console da Questão 2:*

SUBSTITUIR ESSE TEXTO QUE ESTÁ EM LARANJA PELA A SAÍDA DO CONSOLE DO EXERCÍCIO 2  
NÃO ESQUECER DE CUMPRIR AS EXIGÊNCIAS

SERÁ ACEITO SOMETE **SAÍDAS DO CONSOLE NO FORMATO IMAGEM** (NADA DE TEXTO AQUI! ZERA  
ESSA PARTE DA QUESTÃO!)

## QUESTÃO 3 de 4 - Conteúdo até aula 05

**Enunciado:** Você foi contratado para desenvolver um sistema de Venda de uma Empresa Y que vende toras de árvore para outras empresas que vendem madeira. Você ficou com a parte de desenvolver a interface com o cliente.

A Empresa Y opera as vendas da seguinte maneira:

- Tora de Pinho (PIN), o valor do metro cúbico ( $m^3$ ) é de cento e cinquenta reais e quarenta centavos;
- Tora de Peroba (PER), o valor do metro cúbico ( $m^3$ ) é de cento e setenta reais e vinte centavos;
- Tora de Mogno (MOG), o valor do metro cúbico ( $m^3$ ) é de cento e noventa reais e noventa centavos;
- Tora de Ipê (IPE), o valor do metro cúbico ( $m^3$ ) é de duzentos e dez reais e dez centavos;
- Tora de Imbuia (IMB), o valor do metro cúbico ( $m^3$ ) é de duzentos e vinte reais e setenta centavos;
  
- Se a quantidade (em  $m^3$ ) de toras for **menor** que 100 não há desconto na venda (0/100);
- Se a quantidade (em  $m^3$ ) de toras for **igual ou maior** que 100 e **menor** que 500, o desconto será de 4% (4/100);
- Se a quantidade (em  $m^3$ ) de toras for **igual ou maior** que 500 e **menor** que 1000, o desconto será de 9% (9/100);
- Se a quantidade (em  $m^3$ ) de toras for **igual ou maior** que 1000 e **menor ou igual** que 2000, o desconto será de 16% (16/100);
- Se a quantidade (em  $m^3$ ) de toras for **maior** que 2000, não é aceito pedidos com essa quantidade de toras;
  
- ♦ Para o **adicional** de transporte rodoviário (1) é cobrado um valor **extra** de 1000 reais;
- ♦ Para o **adicional** de transporte ferroviário (2) é cobrado um valor **extra** de 2000 reais;
- ♦ Para o **adicional** de transporte hidroviário (3) é cobrado um valor **extra** de 2500 reais;

O valor final da conta é calculado da seguinte maneira:

$$\text{total} = ((\text{tipoMadeira} * \text{qtdToras}) * (1 - \text{desconto})) + \text{transporte}$$

**Elabore um programa em Python que:**

- Deve-se implementar o print com uma mensagem de boas-vindas que apareça o seu **nome e sobrenome** (somente print, não usar input aqui). **[EXIGÊNCIA DE CÓDIGO 1 de 7]**;
- Deve-se implementar a função **escolha\_tipo()** que **não** recebe parâmetros e que: **[EXIGÊNCIA DE CÓDIGO 2 de 7]**
  - Pergunta o **tipo de madeira** desejado;
  - Retorna o VALOR do tipo de madeira** com base na escolha do usuário (use **return**);
  - Repete a pergunta do item **B.a** se digitar uma opção diferente de: PIN/PER/MOG/IPE/IMB;

- C. Deve-se implementar a função **qtd\_toras()** que **não** recebe parâmetros e que: [EXIGÊNCIA DE CÓDIGO 3 de 7];
- Pergunta a **quantidade de toras**;
  - Retorna** (use **return**) a **quantidade de toras E o valor do desconto** (os dois valores) seguindo a regra do enunciado;
  - Repete a pergunta do item **C.a** se digitar um valor acima de 2000 ou valor não numérico (use try/except para não numérico)
- D. Deve-se implementar a função **transporte()** que **não** recebe parâmetros e que: [EXIGÊNCIA DE CÓDIGO 4 de 7];
- Pergunta pelo serviço **adicional de transporte**;
  - Retorna** (use **return**) o **valor** de apenas uma das **opções** de **transporte**;
  - Repetir a pergunta item **D.a** se digitar uma opção diferente de: 1/2/3;
- E. Deve-se implementar o total a pagar no código principal (**main**), ou seja, não pode estar dentro de função, conforme o enunciado [EXIGÊNCIA DE CÓDIGO 5 de 7];
- F. Deve-se implementar **try/except** [EXIGÊNCIA DE CÓDIGO 6 de 7];
- G. Deve-se inserir comentários relevantes no código [EXIGÊNCIA DE CÓDIGO 7 de 7];

#### Teste seu código atendendo as seguintes exigências:

- H. Deve-se apresentar na saída de console uma mensagem com o seu **nome** e **sobrenome** [EXIGÊNCIA DE SAÍDA DE CONSOLE 1 de 4];
- I. Deve-se apresentar na saída de console um pedido no qual o usuário errou a opção de tipo de madeira [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 4];
- J. Deve-se apresentar na saída de console um pedido no qual o usuário digitou um valor que ultrapasse a quantidade máxima de toras aceitas (2000) [EXIGÊNCIA DE SAÍDA DE CONSOLE 3 de 4];
- K. Deve-se apresentar na saída de console um pedido com opção de tipo de madeira, quantidade de toras e transporte válidos [EXIGÊNCIA DE SAÍDA DE CONSOLE 4 de 4];

## EXEMPLO DE SAÍDA DE CONSOLE:

```
➡ Bem vindo a Madeireira do Lenhador Bruno Kostiuk Nome completo

Entre com o Tipo de Madeira desejado
PIN - Tora de Pinho
PER - Tora de Peroba
MOG - Tora de Mogno
IPE - Tora de Ipê
IMB - Tora de Imbuia Errou o tipo de Madeira
>>TÁBUA
Escolha inválida, entre com o modelo novamente

Entre com o Tipo de Madeira desejado
PIN - Tora de Pinho
PER - Tora de Peroba
MOG - Tora de Mogno
IPE - Tora de Ipê
IMB - Tora de Imbuia
>>IPE Errou a quantidade de toras
Entre com a quantidade de toras (m³): 500000
Não aceitamos pedidos com essa quantidade de toras.
Por favor, entre com a quantidade novamente.

Entre com a quantidade de toras (m³): 500

Escolha o tipo de Transporte:
1 - Transporte Rodoviário - R$ 1000.00
2 - Transporte Ferroviário - R$ 2000.00
3 - Transporte Hidroviário - R$ 2500.00
>>3
Total: R$ 98095.50 Pedido com tipo de tora, quantidade de tora e
transporte válidos
```

**Figura 3: Exemplo de saída de console que o aluno deve fazer. Em que se pergunta pelo tipo de tora e se erra opção inicialmente, e que se passa a quantidade de toras acima do aceito. Na sequência, o usuário digitou um tipo de tora, quantidade de toras e transporte válidos.**

## *Apresentação de Código da Questão 3:*

SUBSTITUIR ESSE TEXTO QUE ESTÁ EM VERMELHO PELO SEU CÓDIGO DO EXERCÍCIO 3.

NÃO ESQUECER DE CUMPRIR AS EXIGÊNCIAS DE CÓDIGO!!

O CÓDIGO DEVE ESTAR IDENTADO!! (SE NÃO ESTIVER, ZERA ESSA PARTE DA QUESTÃO!)

SERÃO ACEITOS SOMENTE CÓDIGOS NO FORMATO TEXTO (NADA DE IMAGEM NEM PRINT, ZERA ESSA PARTE DA QUESTÃO!).

## *Apresentação de Saída do Console da Questão 3:*

SUBSTITUIR ESSE TEXTO QUE ESTÁ EM LARANJA PELA A SAÍDA DO CONSOLE DO EXERCÍCIO 3, SE NECESSÁRIO, É PERMITIDO INSERIR VÁRIAS IMAGENS.

NÃO ESQUECER DE CUMPRIR AS EXIGÊNCIAS

SERÁ ACEITO SOMENTE **SAÍDAS DO CONSOLE NO FORMATO IMAGEM** (NADA DE TEXTO AQUI! ZERA ESSA PARTE DA QUESTÃO!)

## QUESTÃO 4 de 4 - Conteúdo até aula 06

**Enunciado:** Você e sua equipe de programadores foram contratados por uma pequena empresa para desenvolver um software de gerenciamento de Contatos Comerciais. Este software deve ter o seguinte menu e opções:

- 1) Cadastrar Contato
- 2) Consultar Contato
  1. Consultar Todos
  2. Consultar por Id
  3. Consultar por Atividade
  4. Retornar ao menu
- 3) Remover Contato
- 4) Encerrar Programa

**Elabore um programa em Python que:**

- A. Deve-se implementar o print com uma mensagem de boas-vindas que apareça o seu **nome e sobrenome** (somente print, não usar input aqui) [EXIGÊNCIA DE CÓDIGO 1 de 8];
- B. Deve-se implementar uma lista com o nome de **lista\_contatos** e a variável **id\_global** com valor igual ao número de seu RU [EXIGÊNCIA DE CÓDIGO 2 de 8];
- C. Deve-se implementar uma função chamada **cadastrar\_contato(id)** que recebe **apenas id** como parâmetro e que: [EXIGÊNCIA DE CÓDIGO 3 de 8];
  - a. Pergunta **nome, atividade, telefone** do contato;
  - b. Armazena o **id** (este é fornecido via parâmetro da função), **nome, atividade, telefone** dentro de um dicionário;
  - c. **Copiar** o dicionário para dentro da **lista\_contatos** (utilizar o **copy**);
- D. Deve-se implementar uma função chamada **consultar\_contatos()** que **não** recebe parâmetros e que: [EXIGÊNCIA DE CÓDIGO 4 de 8];
  - a. Deve-se perguntar qual opção deseja (1. Consultar Todos / 2. Consultar por Id / 3. Consultar por Setor / 4. Retornar ao menu):
    - i. Se Consultar Todos, apresentar todos os contatos com todos os seus dados cadastrados;
    - ii. Se Consultar por Id, solicitar ao usuário que informe um id, e apresentar o contato **específico** (apenas 1) com todos os seus dados cadastrados;
    - iii. Se Consultar por Atividade, solicitar ao usuário que informe a atividade, e apresentar o(s) contato(s) que exercem aquela atividade com todos os seus dados cadastrados;
    - iv. Se Retornar ao menu, deve-se **retornar** ao menu principal (return);
    - v. Se Entrar com um valor diferente de 1, 2, 3 ou 4, printar "Opção inválida" e repetir a pergunta **D.a.**
    - vi. Enquanto o usuário não escolher a opção 4, o menu consultar contatos deve se repetir.



- E. Deve-se implementar uma função chamada **remover\_contato()** em que: [EXIGÊNCIA DE CÓDIGO 5 de 8];
- Deve-se pergunta pelo **id** do contato a ser removido;
  - Remover o contato da **lista\_contatos**;
  - Se o id fornecido não for de um contato da lista, printar "**Id inválido**" e repetir a pergunta **E.a.**
- F. Deve-se implementar uma estrutura de menu no código principal (**main**), ou seja, **não pode estar dentro de função**, em que: [EXIGÊNCIA DE CÓDIGO 6 de 8];
- Deve-se pergunta qual opção deseja (1. Cadastrar Contato / 2. Consultar Contato / 3. Remover Contato / 4. Encerrar Programa):
    - Se Cadastrar Contato, chamar a função **cadastrar\_contato (id\_global)** e **em seguida, incrementar** em um **id\_global**;
    - Se Consultar Contato, chamar função **consultar\_contato ()**;
    - Se Remover Contato, chamar função **remover\_contato ()**;
    - Se Encerrar Programa, sair do menu (e com isso acabar a execução do código);
    - Se Entrar com um valor diferente de 1, 2, 3 ou 4, printar "Opção inválida" e repetir a pergunta **F.a.**
    - Enquanto o usuário não escolher a opção 4, o menu deve se repetir.
- G. Deve-se implementar uma **lista de dicionários** (uma lista contendo dicionários dentro) [EXIGÊNCIA DE CÓDIGO 7 de 8];
- H. Deve-se inserir comentários relevantes no código [EXIGÊNCIA DE CÓDIGO 8 de 8];

### Teste seu código atendendo as seguintes exigências:

- I. Deve-se apresentar na saída de console um cadastro do **seu contato** da seguinte forma: para **nome** informe seu **nome completo** (não usar apelidos ou abreviações), para **atividade** informar como **estudante**, e para **telefone** informe sua **RU**. [EXIGÊNCIA DE SAÍDA DE CONSOLE 1 de 6];
- J. Deve-se apresentar na saída de console um cadastro de **mais 2** contatos com mesmo tipo de atividade (por exemplo: marceneiro, padeiro, pintor, pedreiro) [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 6];
- K. Deve-se apresentar na saída de console uma consulta de todos os contatos [EXIGÊNCIA DE SAÍDA DE CONSOLE 3 de 6];
- L. Deve-se apresentar na saída de console uma consulta por código (id) de um dos contatos [EXIGÊNCIA DE SAÍDA DE CONSOLE 4 de 6];
- M. Deve-se apresentar na saída de console uma consulta por atividade em que **2** contatos exerçam a mesma atividade [EXIGÊNCIA DE SAÍDA DE CONSOLE 5 de 6];
- N. Deve-se apresentar na saída de console uma remoção de um dos contatos e em seguida de uma consulta de todos os contatos, provando que o contato foi removido [EXIGÊNCIA DE SAÍDA DE CONSOLE 6 de 6];

## EXEMPLO DE SAÍDA DE CONSOLE:

```
➡ Bem vindo a Lista de Contatos do Bruno Kostiuk Nome completo
-----
----- MENU PRINCIPAL -----
Escolha a opção desejada:
1 - Cadastrar Contato
2 - Consultar Contato(s)
3 - Remover Contato
4 - Sair
>>1
-----
----- MENU CADASTRAR CONTATO -----
Id do Contato: 4297914
Por favor entre com o nome do Contato: Bruno Kostiuk
Por favor entre com a Atividade do contato: Estudante
Por favor entre com o telefone do contato: 4297913
-----
```

Cadastro do primeiro contato,  
com seu nome completo,  
atividade estudante e telefone  
igual ao seu RU

Figura 4.1: Exemplo de saída de console que o aluno deve fazer. Apresenta o print com seu nome completo e é realizado o cadastro do primeiro contato, note que o ID do contato não inicia em 1, pois ele deve iniciar com o seu RU (caso o RU informado não seja o seu, irá receber zero em toda questão). O primeiro contato deve ser cadastrado com SEU NOME COMPLETO, em Atividade informe Estudante e em Contato informe o SEU RU.

```
➡ -----
----- MENU PRINCIPAL -----
Escolha a opção desejada:
1 - Cadastrar Contato
2 - Consultar Contato(s)
3 - Remover Contato
4 - Sair
>>1
-----
----- MENU CADASTRAR CONTATO -----
Id do Contato: 4297915
Por favor entre com o nome do Contato: Tamy
Por favor entre com a Atividade do contato: Professor
Por favor entre com o telefone do contato: 99998888
-----
----- MENU PRINCIPAL -----
Escolha a opção desejada:
1 - Cadastrar Contato
2 - Consultar Contato(s)
3 - Remover Contato
4 - Sair
>>1
-----
----- MENU CADASTRAR CONTATO -----
Id do Contato: 4297916
Por favor entre com o nome do Contato: Osmar
Por favor entre com a Atividade do contato: Professor
Por favor entre com o telefone do contato: 88889999
-----
```

Cadastre mais dois contatos com  
mesmo tipo de Atividade

Figura 4.2: Exemplo de saída de console que o aluno deve fazer. São cadastrados mais dois contatos com mesmo tipo de Atividade.

```

➡ ----- MENU CONSULTAR CONTATOS -----
Escolha a opção desejada:
1 - Consultar Todos os Contatos
2 - Consultar Contato por id
3 - Consultar Contato(s) por Atividade
4 - Retornar
>>1
-----
id: 4297914
nome: Bruno KostiuK
atividade: Estudante
telefone: 4297913

id: 4297915
nome: Tamy
atividade: Professor
telefone: 99998888

id: 4297916
nome: Osmar
atividade: Professor
telefone: 88889999
-----

```

**Figura 4.3: Exemplo de saída de console que o aluno deve fazer. Em que se consulta Todos os contatos cadastrados.**

```

➡ Escolha a opção desejada:
1 - Consultar Todos os Contatos
2 - Consultar Contato por id
3 - Consultar Contato(s) por Atividade
4 - Retornar
>>2
Digite o id do contato: 4297914
-----
id: 4297914
nome: Bruno KostiuK
atividade: Estudante
telefone: 4297913

-----

----- MENU CONSULTAR CONTATOS -----
Escolha a opção desejada:
1 - Consultar Todos os Contatos
2 - Consultar Contato por id
3 - Consultar Contato(s) por Atividade
4 - Retornar
>>3
Digite a Atividade do(s) Contato(s): Professor
-----
id: 4297915
nome: Tamy
atividade: Professor
telefone: 99998888

id: 4297916
nome: Osmar
atividade: Professor
telefone: 88889999
-----

```

**Figura 4.4: Exemplo de saída de console que o aluno deve fazer. Em que se consulta o contato com id número 4297914 e consulta pelo nome da Atividade (Professor).**

```
-----  
----- MENU PRINCIPAL -----  
Escolha a opção desejada:  
1 - Cadastrar Contato  
2 - Consultar Contato(s)  
3 - Remover Contato  
4 - Sair  
>>3  
-----  
----- MENU REMOVER CONTATO -----  
Digite o id do contato a ser removido: 4297914  
Contato removido com sucesso!  
-----  
----- MENU PRINCIPAL -----  
Escolha a opção desejada:  
1 - Cadastrar Contato  
2 - Consultar Contato(s)  
3 - Remover Contato  
4 - Sair  
>>2  
-----  
----- MENU CONSULTAR CONTATOS -----  
Escolha a opção desejada:  
1 - Consultar Todos os Contatos  
2 - Consultar Contato por id  
3 - Consultar Contato(s) por Atividade  
4 - Retornar  
>>1  
-----  
id: 4297915  
nome: Tamy  
atividade: Professor  
telefone: 99998888  
-----  
id: 4297916  
nome: Osmar  
atividade: Professor  
telefone: 88889999  
-----
```

**Figura 4.5: Exemplo de saída de console que o aluno deve fazer. Em que se remove o contato de Id número 4297914 e depois se faz uma consulta de todos os contatos.**

## *Apresentação de **Código da Questão 4:***

SUBSTITUIR ESSE TEXTO QUE ESTÁ EM VERMELHO PELO SEU CÓDIGO DO EXERCÍCIO 4.

NÃO ESQUECER DE CUMPRIR AS EXIGÊNCIAS DE CÓDIGO!!

O CÓDIGO DEVE ESTAR IDENTADO!! (SE NÃO ESTIVER, ZERA ESSA PARTE DA QUESTÃO!)

SERÃO ACEITOS SOMENTE CÓDIGOS NO FORMATO TEXTO (NADA DE IMAGEM NEM PRINT, ZERA ESSA PARTE DA QUESTÃO!).

## *Apresentação de Saída do Console da Questão 4:*

SUBSTITUIR ESSE TEXTO QUE ESTÁ EM LARANJA PELA A SAÍDA DO CONSOLE DO EXERCÍCIO 4, SE NECESSÁRIO, É PERMITIDO INSERIR VÁRIAS IMAGENS.

NÃO ESQUECER DE CUMPRIR AS EXIGÊNCIAS

SERÁ ACEITO SOMENTE **SAÍDAS DO CONSOLE NO FORMATO IMAGEM** (NADA DE TEXTO AQUI! ZERA ESSA PARTE DA QUESTÃO!)

# CRITÉRIOS DE AVALIAÇÃO

SOMENTE INFORMATIVO. PODE APAGAR AO ENTREGAR O TRABALHO.

A avaliação se dará através de **critérios estritamente objetivos**:

- Dentro de cada questão, metade da nota se refere as exigências de implementação do código (EC), e a outra metade as exigências de saída (ES) de funcionamento do código:

$$NOTA_{QUESTÃO} = \frac{NOTA_{EC} + NOTA_{ES}}{2}$$

- Cada questão vale um quarto da nota da atividade. Assim, calculamos a sua nota final na atividade prática com a fórmula abaixo:

$$NOTA_{AP} = \frac{NOTA_{QUESTÃO_1} + NOTA_{QUESTÃO_2} + NOTA_{QUESTÃO_3} + NOTA_{QUESTÃO_4}}{4}$$

Os critérios e suas respectivas pontuações estão na tabela a seguir:

QUESTÃO 1		
Exigências de Código (EC)		Pts
EC 1	Implementar corretamente a mensagem de Boas-vindas com nome e sobrenome	30
EC 2	Implementar corretamente o input de valorBase e idade	10
EC 3	Implementar corretamente a regra de cobrança mensal conforme o enunciado	15
EC 4	Implementar corretamente o valor mensal	15
EC 5	Implementar corretamente o if, elif e else	20
EC 6	Inserção de comentários no código	10
Total		100
Exigências de Saída (ES)		Pts
ES 1	Apresentar o print de Boas-vindas com nome e sobrenome	30
ES 2	Informar no teste uma idade maior ou igual a 29 anos	35
ES 2	Apresentar o Valor Mensal	35
Total		100





<b>QUESTÃO 2</b>		
<b>Exigências de Código (EC)</b>		<b>Pts</b>
EC 1	Implementar corretamente a mensagem de Boas-vindas com nome e sobrenome	15
EC 1	Apresentar o Menu para o cliente	15
EC 2	Implementar corretamente o input do sabor e print de sabor invalido	10
EC 3	Implementar corretamente o input do tamanho e print de tamanho invalido	10
EC 4	Implementar corretamente as regras para combinação de sabores e tamanhos	10
EC 5	Implementar corretamente o acumulador com a soma dos valores	10
EC 6	Implementar corretamente a condição de "Deseja pedir mais alguma coisa?"	10
EC 7	Implementar corretamente o while, continue e break	10
EC 8	Inserção de comentários no código	10
<b>Total</b>		<b>100</b>
<b>Exigências de Saída (ES)</b>		<b>Pts</b>
ES 1	Apresentar o print de Boas-vindas com nome e sobrenome	15
ES 1	Apresentar o menu de opções	15
ES 2	Apresentar um pedido em que o usuário errou ao digitar o sabor	20
ES 3	Apresentar um pedido em que o usuário errou ao digitar o tamanho	20
ES 4	Apresentar um pedido com dois sabores e dois tamanhos diferentes	30
<b>Total</b>		<b>100</b>

<b>QUESTÃO 3</b>		
<b>Exigências de Código (EC)</b>		<b>Pts</b>
EC 1	Implementar corretamente a mensagem de Boas-vindas com nome e sobrenome	30
EC 2	Implementar corretamente a função escolha_tipo()	10
EC 3	Implementar corretamente a função qtd_toras()	20
EC 4	Implementar corretamente a função transporte()	10
EC 5	Implementar corretamente o total a pagar conforme a regra do enunciado	10
EC 6	Implementar corretamente a cláusula try/except	10
EC 7	Inserção de comentários no código	10
<b>Total</b>		<b>100</b>
<b>Exigências de Saída (ES)</b>		<b>Pts</b>
ES 1	Apresentar o print de Boas-vindas com nome e sobrenome	30
ES 2	Apresentar um pedido em que o usuário errou a opção de tipo de madeira	20
ES 3	Apresentar um pedido em que o usuário ultrapassou o número máximo de toras	20
ES 4	Apresentar um pedido válido.	30
<b>Total</b>		<b>100</b>

<b>QUESTÃO 4</b>		
<b>Exigências de Código (EC)</b>		<b>Pts</b>
EC 1	Implementar corretamente a mensagem de Boas-vindas com nome e sobrenome	30
EC 2	Implementar corretamente lista de contatos e id_global	5
EC 3	Implementar corretamente a função cadastrar_contato(id)	10
EC 4	Implementar corretamente a função consultar_contato() opção 1	5
EC 4	Implementar corretamente a função consultar_contato() opção 2	5
EC 4	Implementar corretamente a função consultar_contato() opção 3	5
EC 4	Implementar corretamente a função consultar_contato() opção 4	5
EC 5	Implementar corretamente a função remover_contato()	10
EC 6	Implementar corretamente o menu principal conforme enunciado	10
EC 7	Implementar corretamente uma lista contendo dicionários	5
EC 8	Inserção de comentários no código	10
<b>Total</b>		<b>100</b>
<b>Exigências de Saída (ES)</b>		<b>Pts</b>
ES 1	Apresentar o cadastro do seu contato (identificação pessoal)	30
ES 2	Apresentar o cadastro de 2 contatos com mesma atividade	15
ES 3	Apresentar uma consulta de todos os contatos	15
ES 4	Apresentar uma consulta por código (id)	15
ES 5	Apresentar uma consulta por atividade com 2 contatos da mesma atividade	15
ES 6	Apresentar uma remoção seguida de uma consulta de todos os contatos	10
<b>Total</b>		<b>100</b>