

Linguagem de programação, lista 2, 2º ADS noturno, Vinícius Schutt de Almeida

**Exercício 1:** Escreva um programa que leia a idade de 10 pessoas e armazene-as em uma lista. Calcule e mostre:

- a) a menor idade
- b) a média das idades
- c) a quantidade de pessoas que tem idade entre 20 e 30 anos (inclusive)
- d) a quantidade de pessoas com idade maior que a média

**Código**

```
idades=[] #lista com as idades
menoridade=0

vintetrinta=[] #lista para armazenar as pessoas com idades entre 20 e 30 anos
maiorqmedia=[] #lista para armazenar as pessoas com idade acima da media do intervalo
x=0 #contador inicializado em 0
leridade=[] #lista que recebe as idades e depois transfere para a lista "idades" principal

while x<10:
    leridade=int(input("Informe as idades dessas 10 pessoas: "))
    idades.append(leridade)

    if leridade>19 and leridade<31: #c verifica se a idade de cada um dos pontos da lista está
    entre o intervalo 20~30
        vintetrinta.append(leridade)

    x+=1

menoridade=min(idades) #a identifica a menor idade
media=sum(idades)/len(idades) #b calcula a media das idades
a=0

while a<10:
    if idades[a] > media: #d identifica se a idade de cada um dos pontos da lista é maior que a
    media ou nao
        b=idades[a]
        maiorqmedia.append(b)

    a+=1
```

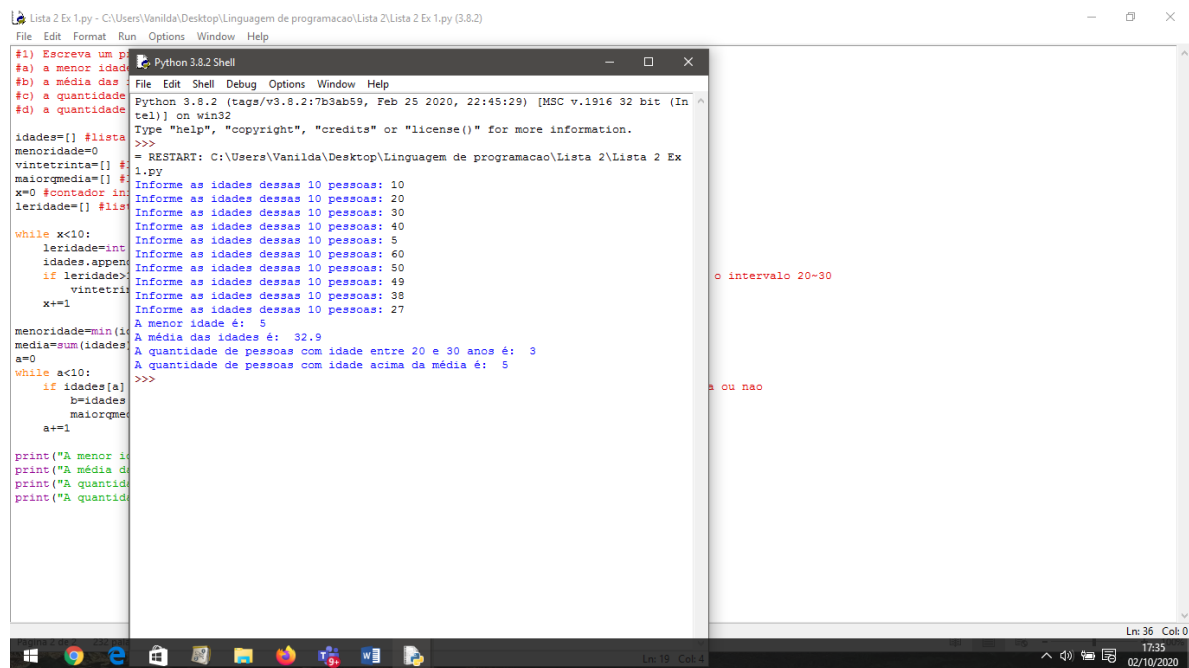
```
print("A menor idade é: ", menoridade)
```

```
print("A média das idades é: ", media)
```

```
print("A quantidade de pessoas com idade entre 20 e 30 anos é: ", len(vintetrinta))
```

```
print("A quantidade de pessoas com idade acima da média é: ", len(maiorqmedia))
```

### Print da tela de execução



```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> = RESTART: C:\Users\Vanilda\Desktop\Linguagem de programacao\Lista 2\Lista 2 Ex 1.py
Informe as idades dessas 10 pessoas: 10
Informe as idades dessas 10 pessoas: 20
Informe as idades dessas 10 pessoas: 30
Informe as idades dessas 10 pessoas: 40
Informe as idades dessas 10 pessoas: 5
Informe as idades dessas 10 pessoas: 60
Informe as idades dessas 10 pessoas: 50
Informe as idades dessas 10 pessoas: 49
Informe as idades dessas 10 pessoas: 38
Informe as idades dessas 10 pessoas: 27
A menor idade é: 5
A média das idades é: 32.9
A quantidade de pessoas com idade entre 20 e 30 anos é: 3
A quantidade de pessoas com idade acima da média é: 5
>>>
```

**Exercício 2:** Faça um programa que preencha uma lista com 10 cores diferentes. Depois permita fazer uma pesquisa se uma determinada cor existe armazenada na lista, se existir deve ser impresso na tela a cor e em qual posição (índice) esta cor está armazenada. A pesquisa deve ser feita até que seja digitado FIM na cor a ser pesquisada na lista.

### Código:

```
cor=['vermelho','azul','amarelo','verde','roxo','branco','preto','laranja','bege','marrom']
```

```
n=1
```

```
print('Busque pelas cores na lista (digite FIM para encerrar)')
```

```
while n>0:
```

```
    n=1
```

```
    pesq = str(input('Busca de cores: '))
```

```
    if pesq in cor:
```

```
        print('A cor ', pesq, 'está presente na posição ',cor.index(pesq))
```

```
    if pesq == "FIM":
```

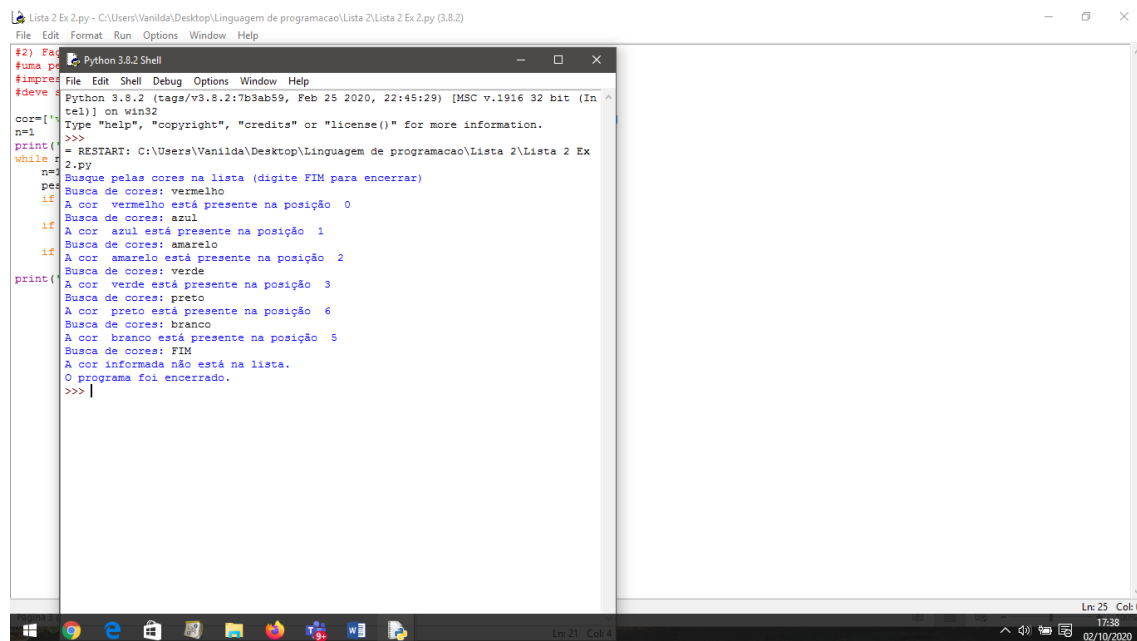
```
n=0
```

```
if pesq not in cor:
```

```
    print('A cor informada não está na lista.')
```

```
print('O programa foi encerrado.')
```

### Print da tela de execução



**Exercício 3:** Faça um programa que preencha duas listas, lista A e lista B com 5 números em cada. Gere a lista C, com os números da lista A e lista B. Depois calcule e mostre na tela a quantidade de números perfeitos. Um número é perfeito quando ele é igual à soma dos seus divisores excetuando ele próprio. (Exemplo: 6 é perfeito,  $6 = 1 + 2 + 3$ , que são seus divisores).

### Código:

```
listaA=[] #sem disclaimer
```

```
listaB=[] #//
```

```
listaC=[] #//
```

```
nperf=[]#números perfeitos
```

```
a=0
```

```
b=0
```

```
print("Cada lista será composta por cinco números. Informe os números da lista A: ")
```

```
while a<5:
```

```
    nsLA=int(input('Número: ')) #numeros lista A
```

```

listaA.append(nsLA)

listaC.append(nsLA)

a+=1


print("Agora, informe os números para a lista B: ")

while b<5:

    nsLB=int(input('Número: ')) #numeros lista B

    listaB.append(nsLB)

    listaC.append(nsLB)

    b+=1

i=0

#d=1

for i in range(0,10):

    soma=0

    d=1

    while d<listaC[i]:

        if listaC[i]%d==0:

            soma += d

            if soma==listaC[i]:

                nperf.append(listaC[i])

            d+=1

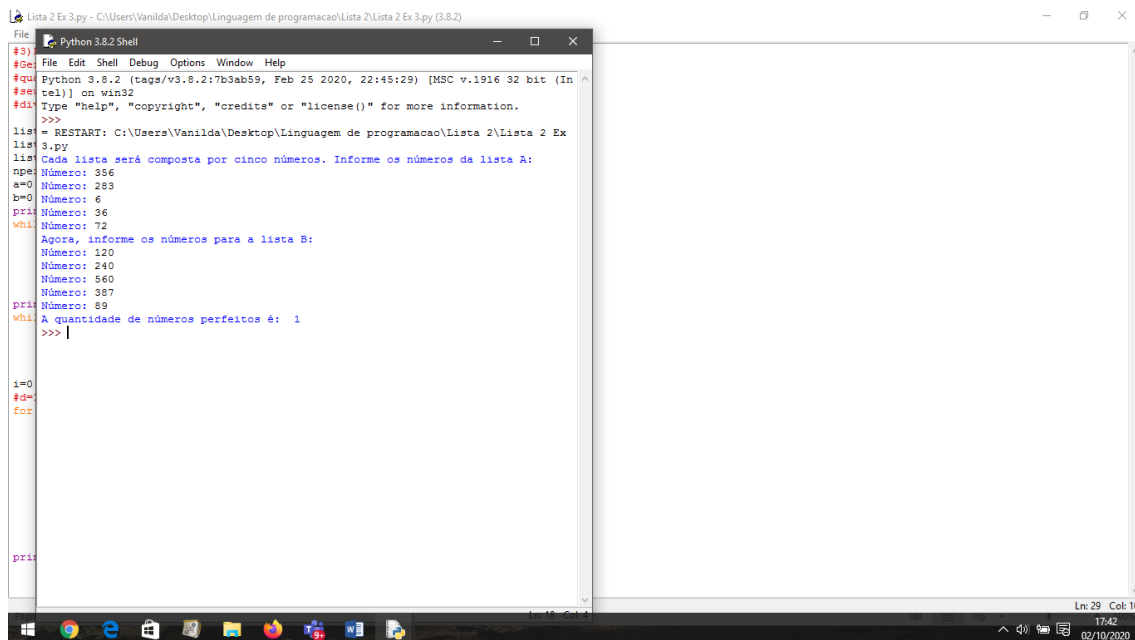
        if d==listaC[i]:

            i+=1

print("A quantidade de números perfeitos é: ", len(nperf))

```

**Print da tela de execução**



```
#3)
#Ge Python 3.8.2 Shell
#qu Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (In
#se tel)] on win32
#di Type "help", "copyright", "credits" or "license()" for more information.
>>>
lis = RESTART: C:\Users\Vanilda\Desktop\Linguagem de programacao\Lista 2\Lista 2 Ex
lis 3.py
lis Cada lista será composta por cinco números. Informe os números da lista A:
npe1 Número: 356
a=0 Número: 283
b=0 Número: 6
pri1 Número: 36
whi1 Número: 72
Agora, informe os números para a lista B:
Número: 120
Número: 240
Número: 560
Número: 387
pri1 Número: 89
whi1 A quantidade de números perfeitos é: 1
>>>

i=0
#d=
for

pri1
```

**Exercício 4:** Faça um programa que preencha duas listas com 10 elementos cada. Depois percorra essas duas listas e gere uma terceira com números que se repetem nas duas listas. Mostre as três listas na tela.

**Código:**

```
lista1=[]
```

```
lista2=[]
```

```
lista3=[]
```

```
x=0
```

```
while x < 10:
```

```
    valoreslista1=(input("Próximo número da 1ª lista: ", )) #entradas para 1ª lista
```

```
    lista1.append(valoreslista1)
```

```
    valoreslista2=(input("Próximo número da 2ª lista: ", )) #entradas para 2ª lista
```

```
    lista2.append(valoreslista2)
```

```
    x=x+1
```

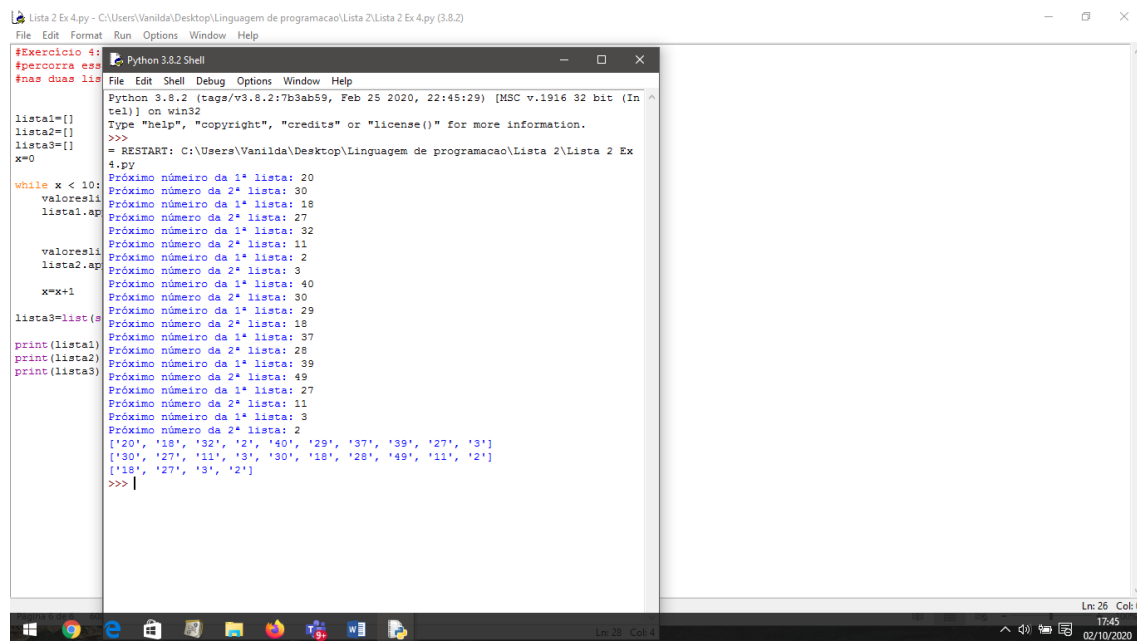
```
lista3=list(set(lista1).intersection(lista2))
```

```
print(lista1)
```

```
print(lista2)
```

```
print(lista3)
```

## Print tela de execução



```
#Exercício 4:
#percorra essas
#nas duas listas

lista1=[]
lista2=[]
lista3=[]
x=0

while x < 10:
    valores1=
    lista1.ap
    valores2=
    lista2.ap
    x=x+1
    lista3=list(
    print(lista1)
    print(lista2)
    print(lista3)

Python 3.8.2 Shell
Python 3.8.2 (tags/v3.8.2:7b9ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\Vanilda\Desktop\Linguagem de programacao\Lista 2\Lista 2 Ex 4.py
Próximo número da 1ª lista: 20
Próximo número da 2ª lista: 30
Próximo número da 1ª lista: 18
Próximo número da 2ª lista: 27
Próximo número da 1ª lista: 32
Próximo número da 2ª lista: 11
Próximo número da 1ª lista: 2
Próximo número da 2ª lista: 3
Próximo número da 1ª lista: 40
Próximo número da 2ª lista: 30
Próximo número da 1ª lista: 29
Próximo número da 2ª lista: 18
Próximo número da 1ª lista: 37
Próximo número da 2ª lista: 28
Próximo número da 1ª lista: 39
Próximo número da 2ª lista: 49
Próximo número da 1ª lista: 27
Próximo número da 2ª lista: 11
Próximo número da 1ª lista: 3
Próximo número da 2ª lista: 2
['20', '18', '32', '2', '40', '29', '37', '39', '27', '3']
['30', '27', '11', '3', '30', '18', '28', '49', '11', '2']
['18', '27', '3', '2']
>>>
```

**Exercício 5:** Faça um programa que preencha uma lista com os nomes de 5 produtos, e outra lista com o valor dos produtos. Calcule e mostre:

- a quantidade de produtos que o valor é abaixo de 10 reais;
- a média dos valores dos produtos;
- a quantidade de produtos que valor acima da média;
- a maior valor e o nome do produto;
- faça uma listagem que imprima na tela (Nome Vlr do produto)

**Código:**

```
nomes=[]
```

```
valores=[]
```

```
abaixode10=[]
```

```
acimadamedia=[]
```

```
nomeevalor=[]
```

```
maiorvalor=[]
```

```
x=0
```

```
while x<5:
```

```
    listanomes=str(input("Digite os nomes dos produtos: "))
```

```

nomes.append(listanomes)
nomeevalor.append(listanomes)
listavalores=float(input("Digite os valores dos produtos: "))
valores.append(listavalores)
nomeevalor.append(listavalores)
if listavalores<10 and listavalores>0: #a
    abaixode10.append(valores)
if listavalores == max(valores): #d
    maiorvalor=listavalores
    prod=listanomes
x+=1

x=0

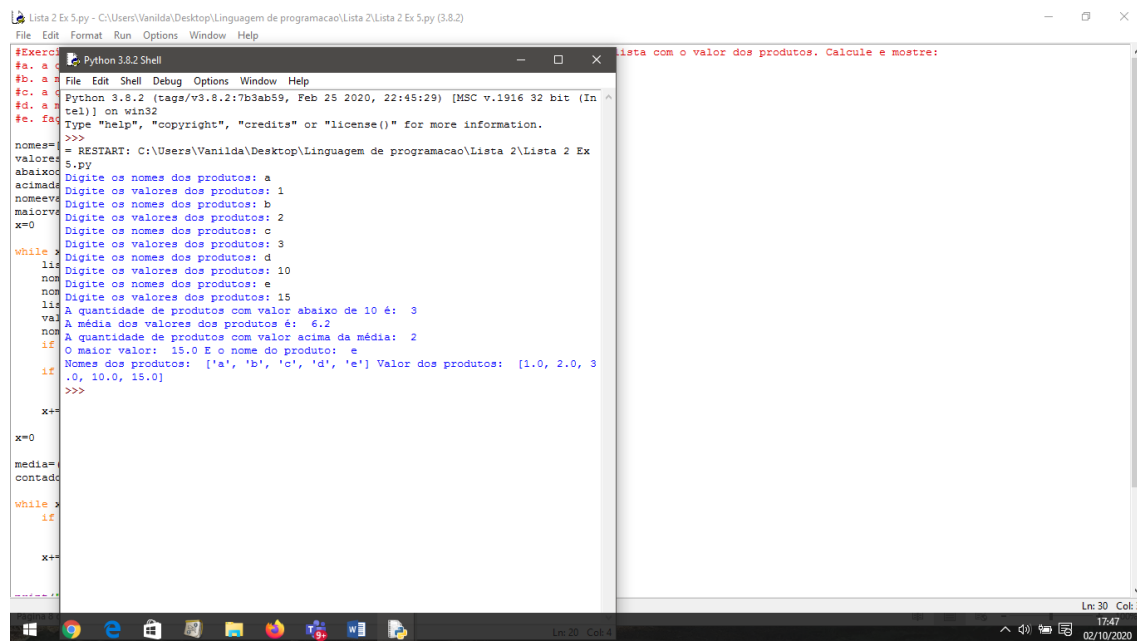
media=(sum(valores)/len(valores)) #b
contador=len(valores)

while x<contador: #c
    if valores[x] > media:
        a=valores[x]
        acimadamedia.append(a)
    x+=1

print("A quantidade de produtos com valor abaixo de 10 é: ", len(abaixode10))
print("A média dos valores dos produtos é: ", media)
print("A quantidade de produtos com valor acima da média: ", len(acimadamedia))
print("O maior valor: ", maiorvalor, "E o nome do produto: ", prod)
print("Nomes dos produtos: ", nomes, "Valor dos produtos: ", valores) #e

```

### Print tela de execução



**Exercício 6:** Faça um programa que percorra duas listas e gere uma terceira lista sem os elementos repetidos. Mostrar na tela as 3 listas.

**Código:**

```
listaA = []
```

```
listaB = []
```

```
listaC = []
```

 $x = 1$ 

```
while x > 0:
```

```
print("Quando quiser encerrar o programa, digite 0.")
```

```
a = int(input("Insira um número na lista A: "))
```

```
if a > 0:
```

```
listaA.append(a)
```

```
listaC.append(a)
```

```
else:
```

 $x = 0$  $x = 1$ 

```
while x > 0:
```

```
print("Quando quiser encerrar o programa, digite 0.")
```



```
b = int(input("Insira um número na lista B: "))
```

```
if b > 0:
```

```
    listaB.append(b)
```

```
    listaC.append(b)
```

```
else:
```

```
    x = 0
```

```
def remove_repetidos(listaC):
```

```
    l=[]
```

```
    for i in listaC:
```

```
        if i not in l:
```

```
            l.append(i)
```

```
    l.sort()
```

```
    return l
```

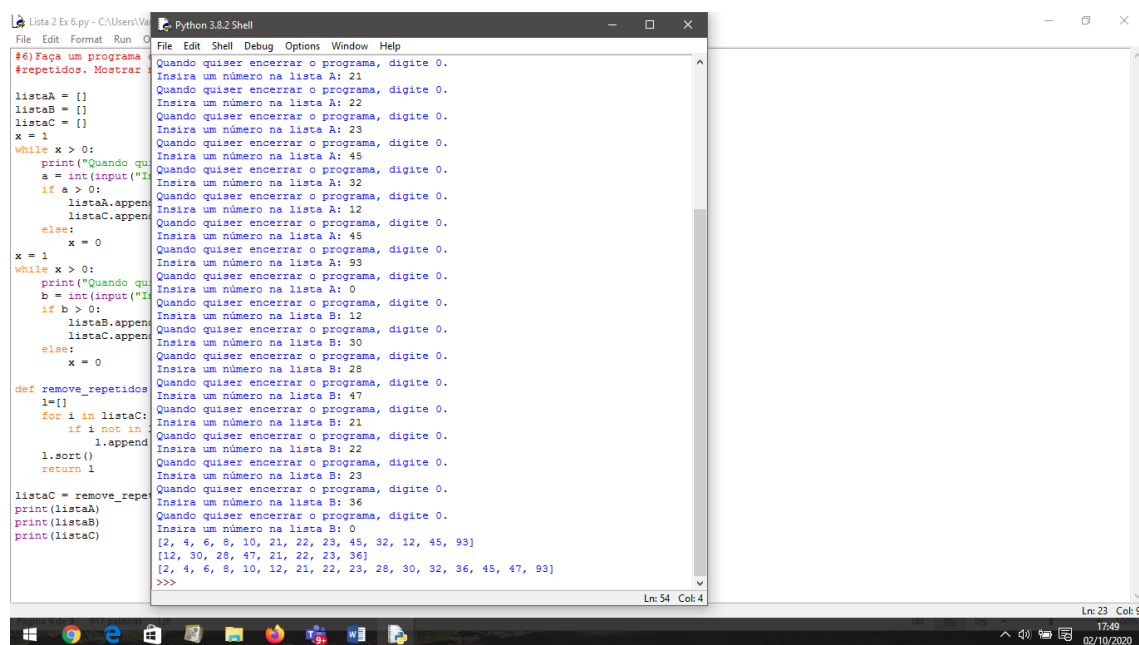
```
listaC = remove_repetidos(listaC)
```

```
print(listaA)
```

```
print(listaB)
```

```
print(listaC)
```

### Print da tela de execução



```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
#6) Faça um programa
#repetidos. Mostre:

listaA = []
listaB = []
listaC = []
x = 1
while x > 0:
    print("Quando quiser encerrar o programa, digite 0.")
    a = int(input("Insira um número na lista A: "))
    if a > 0:
        listaA.append(a)
        listaC.append(a)
    else:
        x = 0
x = 1
while x > 0:
    print("Quando quiser encerrar o programa, digite 0.")
    b = int(input("Insira um número na lista B: "))
    if b > 0:
        listaB.append(b)
        listaC.append(b)
    else:
        x = 0
def remove_repetidos(listaC):
    l=[]
    for i in listaC:
        if i not in l:
            l.append(i)
    l.sort()
    return l
listaC = remove_repetidos(listaC)
print(listaA)
print(listaB)
print(listaC)

Quando quiser encerrar o programa, digite 0.
Insira um número na lista A: 21
Quando quiser encerrar o programa, digite 0.
Insira um número na lista A: 22
Quando quiser encerrar o programa, digite 0.
Insira um número na lista A: 23
Quando quiser encerrar o programa, digite 0.
Insira um número na lista A: 45
Quando quiser encerrar o programa, digite 0.
Insira um número na lista A: 32
Quando quiser encerrar o programa, digite 0.
Insira um número na lista A: 12
Quando quiser encerrar o programa, digite 0.
Insira um número na lista A: 45
Quando quiser encerrar o programa, digite 0.
Insira um número na lista A: 93
Quando quiser encerrar o programa, digite 0.
Insira um número na lista A: 0
Quando quiser encerrar o programa, digite 0.
Insira um número na lista B: 12
Quando quiser encerrar o programa, digite 0.
Insira um número na lista B: 30
Quando quiser encerrar o programa, digite 0.
Insira um número na lista B: 28
Quando quiser encerrar o programa, digite 0.
Insira um número na lista B: 47
Quando quiser encerrar o programa, digite 0.
Insira um número na lista B: 21
Quando quiser encerrar o programa, digite 0.
Insira um número na lista B: 22
Quando quiser encerrar o programa, digite 0.
Insira um número na lista B: 23
Quando quiser encerrar o programa, digite 0.
Insira um número na lista B: 36
Quando quiser encerrar o programa, digite 0.
Insira um número na lista B: 0
[2, 4, 6, 8, 10, 21, 22, 23, 45, 32, 12, 45, 93]
[12, 30, 28, 47, 21, 22, 23, 36]
[2, 4, 6, 8, 10, 12, 21, 22, 23, 28, 30, 32, 36, 45, 47, 93]
>>>
```

**Exercício 7:** Escreva um programa que gere uma lista que é resultado do produto de duas listas L1 e L2. Mostre na tela as 3 listas.

**Código:**

```
lista1=[]

lista2=[]

x=1

for i in range(3):

    lista1.append(int(input('Insira um número na lista 1: ')))

    lista2.append(int(input('Insira um número na lista 2: ')))

C=[]

for i in range(3):

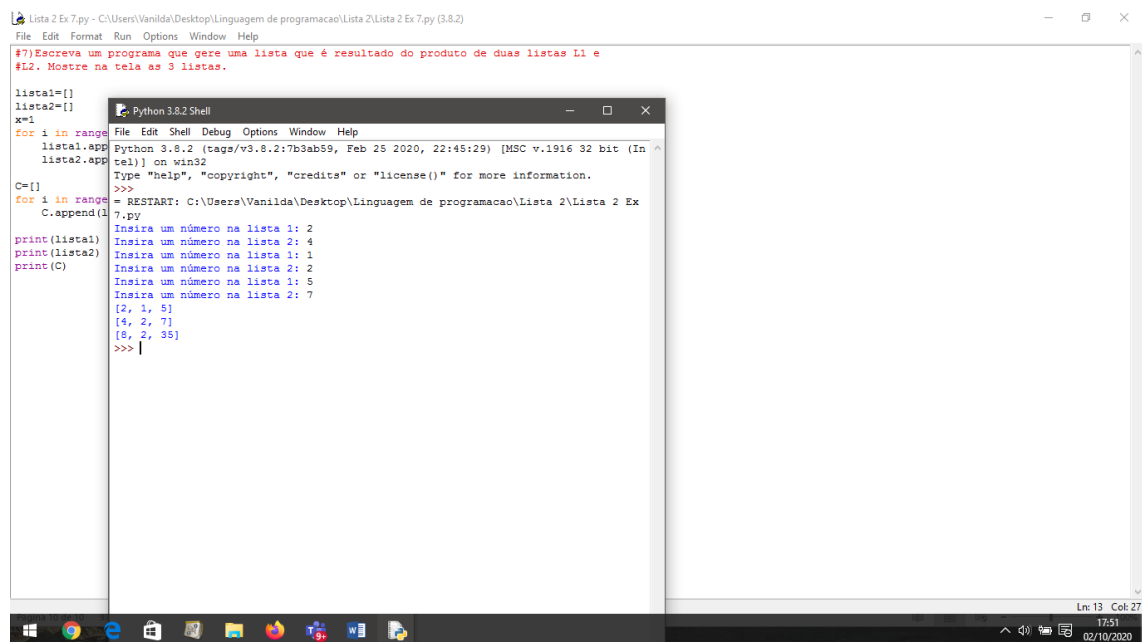
    C.append(lista1[i]*lista2[i])

print(lista1)

print(lista2)

print(C)
```

**Print da tela de execução**



```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (In
tel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
RESTART: C:\Users\Vanilda\Desktop\Linguagem de programacao\Lista 2\Lista 2 Ex
7.py
Insira um número na lista 1: 2
Insira um número na lista 2: 4
Insira um número na lista 1: 1
Insira um número na lista 2: 2
Insira um número na lista 1: 5
Insira um número na lista 2: 7
[2, 1, 5]
[4, 2, 7]
[8, 2, 35]
>>> |
```