



UNIVERSIDAD JUÁREZ AUTÓNOMA DE TABASCO



División Académica De Ingeniería y Arquitectura

LICENCIATURA:

INGENIERIA ELECTRICA Y ELECTRONICA

ASIGNATURA:

ESTRUCTURA Y ALGORITMO DE DATOS

EXAMEN DE 2DO PARCIAL:

5 EJERCICIOS

DOCENTE:

PROF. MARIA ELENA GARCIA ULIN

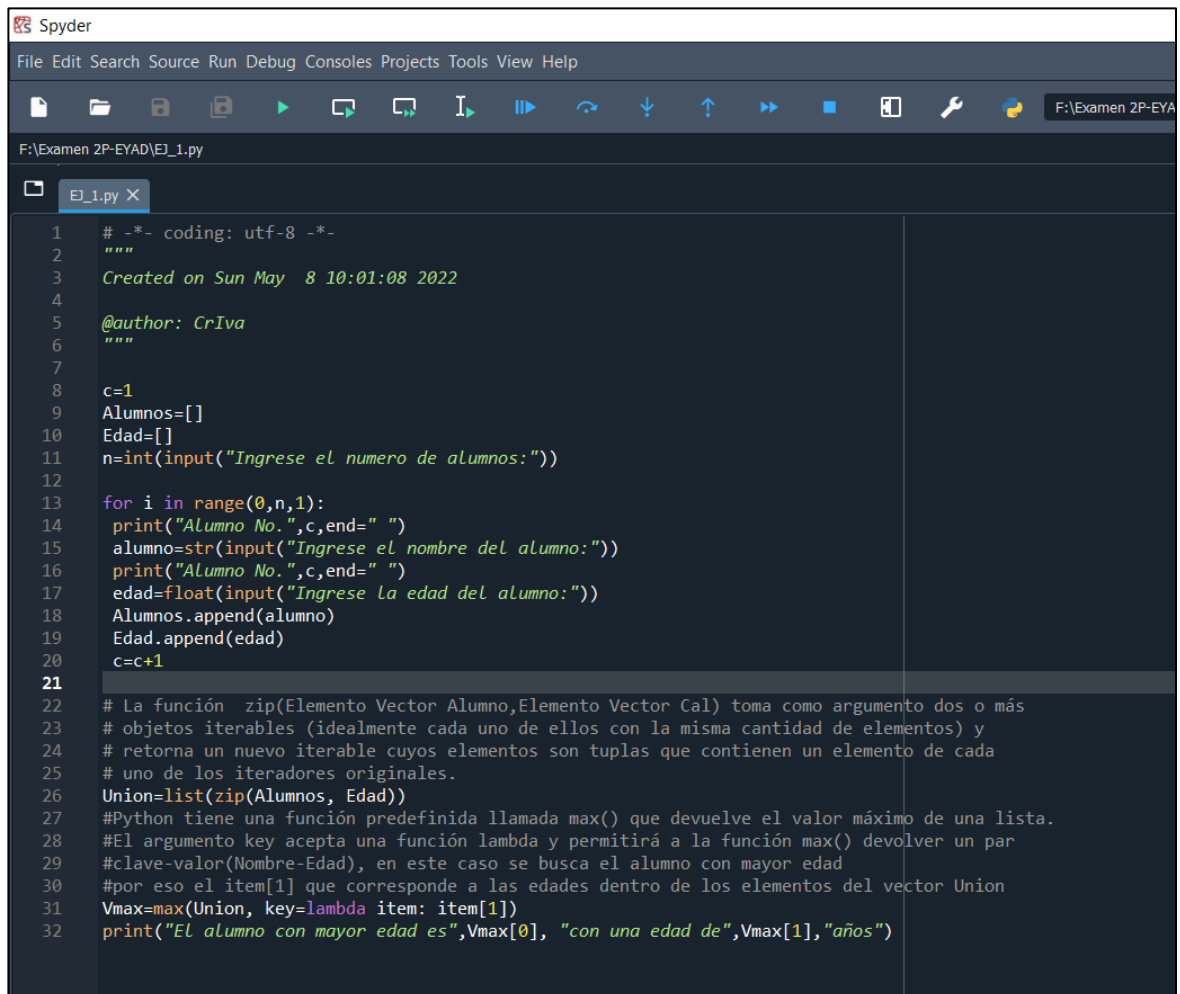
ALUMNO QUE ENTREGA:

192D24008 IVÁN GARCÍA MÉNDEZ

GRUPO:

E12

1. Lea los nombres y las edades de diez alumnos, y que los datos se almacenen en dos vectores, y con base en esto se determine el **nombre del alumno con la edad mayor** del arreglo.



```
1  # -*- coding: utf-8 -*-
2  """
3  Created on Sun May  8 10:01:08 2022
4
5  @author: CrIva
6  """
7
8  c=1
9  Alumnos=[]
10 Edad=[]
11 n=int(input("Ingrese el numero de alumnos:"))
12
13 for i in range(0,n,1):
14     print("Alumno No.",c,end=" ")
15     alumno=str(input("Ingrese el nombre del alumno:"))
16     print("Alumno No.",c,end=" ")
17     edad=float(input("Ingrese la edad del alumno:"))
18     Alumnos.append(alumno)
19     Edad.append(edad)
20     c=c+1
21
22 # La función zip(Elemento Vector Alumno,Elemento Vector Cal) toma como argumento dos o más
23 # objetos iterables (idealmente cada uno de ellos con la misma cantidad de elementos) y
24 # retorna un nuevo iterable cuyos elementos son tuplas que contienen un elemento de cada
25 # uno de los iteradores originales.
26 Union=list(zip(Alumnos, Edad))
27 #Python tiene una función predefinida llamada max() que devuelve el valor máximo de una lista.
28 #El argumento key acepta una función lambda y permitirá a la función max() devolver un par
29 #clave-valor(Nombre-Edad), en este caso se busca el alumno con mayor edad
30 #por eso el item[1] que corresponde a las edades dentro de los elementos del vector Union
31 Vmax=max(Union, key=lambda item: item[1])
32 print("El alumno con mayor edad es",Vmax[0], "con una edad de",Vmax[1],"años")
```

Captura de pantalla del código correspondiente al problema 1, con algunas especificaciones

Inde	Type	Size	Value
0	float	1	20.0
1	float	1	27.0
2	float	1	21.0
3	float	1	22.0
4	float	1	17.0
5	float	1	19.0
6	float	1	24.0
7	float	1	23.0
8	float	1	22.0
9	float	1	25.0

Vector con la edad de los 10 alumnos

Inde	Type	Size	Value
0	str	4	Ivan
1	str	5	Jorge
2	str	5	Edgar
3	str	5	Oscar
4	str	5	Maria
5	str	9	Francisco
6	str	4	Jose
7	str	8	Mercedez
8	str	8	Catalina
9	str	7	Vicente

Vector con los nombres de los 10 alumnos

```

Help Variable Explorer Plots Files
Console 1/A X
Alumno No. 5
Ingrese la edad del alumno:17
Alumno No. 6
Ingrese el nombre del alumno:Francisco
Alumno No. 6
Ingrese la edad del alumno:19
Alumno No. 7
Ingrese el nombre del alumno:Jose
Alumno No. 7
Ingrese la edad del alumno:24
Alumno No. 8
Ingrese el nombre del alumno:Mercedez
Alumno No. 8
Ingrese la edad del alumno:23
Alumno No. 9
Ingrese el nombre del alumno:Catalina
Alumno No. 9
Ingrese la edad del alumno:22
Alumno No. 10
Ingrese el nombre del alumno:Vicente
Alumno No. 10
Ingrese la edad del alumno:25
El alumno con mayor edad es Jorge con una edad de 27.0 años
  
```

Ventana de resultado, en este caso nos dice que el alumno Jorge es el de mayor edad

A continuación, el código del problema 1:

```

c=1
Alumnos=[]
Edad=[]
n=int(input("Ingrese el numero de alumnos:"))

for i in range(0,n,1):
    print("Alumno No.",c,end=" ")
    alumno=str(input("Ingrese el nombre del alumno:"))
    print("Alumno No.",c,end=" ")
    edad=float(input("Ingrese la edad del alumno:"))
    Alumnos.append(alumno)
    Edad.append(edad)
    c=c+1
Union=list(zip(Alumnos, Edad))
Vmax=max(Union, key=lambda item: item[1])
print("El alumno con mayor edad es",Vmax[0], "con una edad de",Vmax[1],"años")

```

2. Modifique el problema anterior, considerando que el vector tiene N elementos y que este número puede ser impar.

```

1  # -*- coding: utf-8 -*-
2  """
3  Created on Mon May 9 11:50:03 2022
4
5  @author: CrIva
6  """
7
8  c=1
9  Alumnos=[]
10 Edad=[]
11 n=0
12
13
14 n=int(input("Ingrese el numero de alumnos:"))
15 if n%2!=0:
16     for i in range(0,n,1):
17         print("Alumno No.",c,end=" ")
18         alumno=str(input("Ingrese el nombre del alumno:"))
19         print("Alumno No.",c,end=" ")
20         edad=float(input("Ingrese la edad del alumno:"))
21         Alumnos.append(alumno)
22         Edad.append(edad)
23         c=c+1
24
25 else:
26     print("Intenete con otro numero ...")
27 if n%2!=0:
28     Union=list(zip(Alumnos, Edad))
29     Vmax=max(Union, key=lambda item: item[1])
30     print("El alumno con mayor edad es",Vmax[0], "con una edad de",Vmax[1],"años")

```

Captura de pantalla del código correspondiente al problema 2

```
Console 1/A X
Ingrese el nombre del alumno:ivan
Alumno No. 1
Ingrese la edad del alumno:20
Alumno No. 2
Ingrese el nombre del alumno:jorge
Alumno No. 2
Ingrese la edad del alumno:22
Alumno No. 3
Ingrese el nombre del alumno:Rafael
Alumno No. 3
Ingrese la edad del alumno:27
Alumno No. 4
Ingrese el nombre del alumno:Maria
Alumno No. 4
Ingrese la edad del alumno:28
Alumno No. 5
Ingrese el nombre del alumno:Selena
Alumno No. 5
Ingrese la edad del alumno:24
El alumno con mayor edad es Maria con una edad de 28.0 años

In [272]: runfile('F:/Examen 2P-EYAD/EJ_2.py', wdir='F:/Examen 2P-EYAD')

Ingrese el numero de alumnos:4
Intenete con otro numero ...
```

Ventana de resultado, podemos ver dentro del marco amarillo que el programa se ejecutó sin ningún problema siempre y cuando N de elementos es impar($n=5$). En el marco rojo podemos la ejecución del programa, pero con N elementos par($n=4$) lo cual no se ejecutará el programa por dicha condición mencionada.

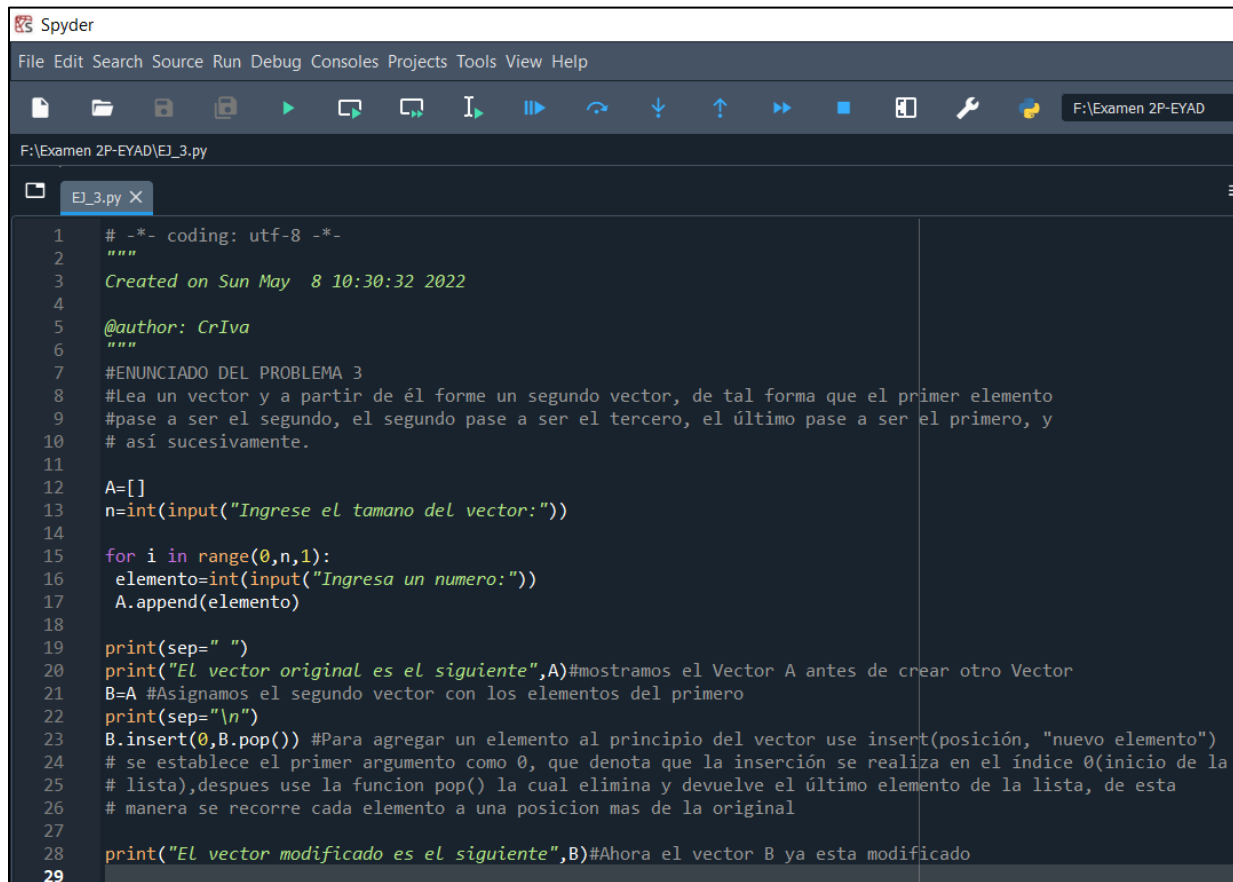
A continuación, el código del problema 2:

```
c=1
Alumnos=[]
Edad=[]

n=int(input("Ingrese el numero de alumnos:"))
if n%2!=0:
    for i in range(0,n,1):
        print("Alumno No.",c,end=" ")
        alumno=str(input("Ingrese el nombre del alumno:"))
        print("Alumno No.",c,end=" ")
        edad=float(input("Ingrese la edad del alumno:"))
        Alumnos.append(alumno)
        Edad.append(edad)
        c=c+1

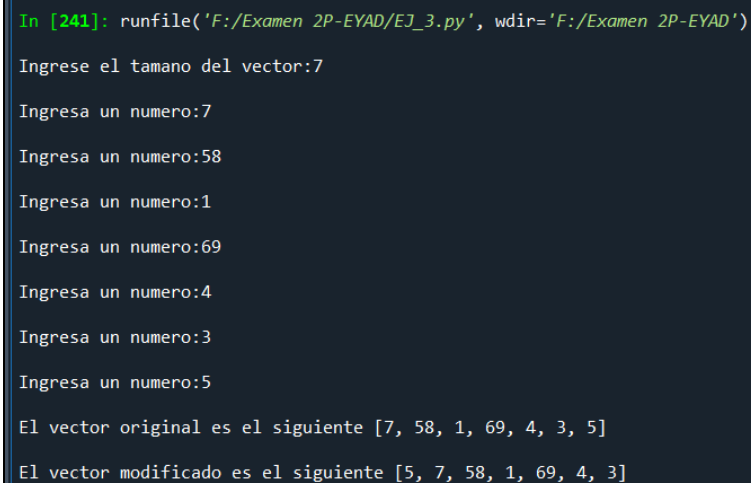
else:
    print("Intenete con otro numero ...")
if n%2!=0:
    Union=list(zip(Alumnos, Edad))
    Vmax=max(Union, key=lambda item: item[1])
    print("El alumno con mayor edad es",Vmax[0], "con una edad de",Vmax[1],"años")
```

3. Lea un vector y a partir de él forme un segundo vector, de tal forma que el primer elemento pase a ser el segundo, el segundo pase a ser el tercero, el último pase a ser el primero, y así sucesivamente.



```
1  #-*- coding: utf-8 -*-
2  """
3  Created on Sun May 8 10:30:32 2022
4
5  @author: CrIva
6  """
7  #ENUNCIADO DEL PROBLEMA 3
8  #Lea un vector y a partir de él forme un segundo vector, de tal forma que el primer elemento
9  #pase a ser el segundo, el segundo pase a ser el tercero, el último pase a ser el primero, y
10 # así sucesivamente.
11
12 A=[]
13 n=int(input("Ingrese el tamaño del vector:"))
14
15 for i in range(0,n,1):
16     elemento=int(input("Ingresa un número:"))
17     A.append(elemento)
18
19 print(sep=" ")
20 print("El vector original es el siguiente",A)#mostramos el Vector A antes de crear otro Vector
21 B=A #Asignamos el segundo vector con los elementos del primero
22 print(sep="\n")
23 B.insert(0,B.pop()) #Para agregar un elemento al principio del vector use insert(posición, "nuevo elemento")
24 # se establece el primer argumento como 0, que denota que la inserción se realiza en el índice 0(inicio de la
25 # lista),despues use la funcion pop() la cual elimina y devuelve el último elemento de la lista, de esta
26 # manera se recorre cada elemento a una posición mas de la original
27
28 print("El vector modificado es el siguiente",B)#Ahora el vector B ya esta modificado
29
```

Captura de pantalla del código correspondiente al problema 3, con algunas especificaciones



```
In [241]: runfile('F:/Examen 2P-EYAD/EJ_3.py', wdir='F:/Examen 2P-EYAD')

Ingrese el tamaño del vector:7

Ingresa un número:7

Ingresa un número:58

Ingresa un número:1

Ingresa un número:69

Ingresa un número:4

Ingresa un número:3

Ingresa un número:5

El vector original es el siguiente [7, 58, 1, 69, 4, 3, 5]

El vector modificado es el siguiente [5, 7, 58, 1, 69, 4, 3]
```

Ventana de resultado, en este caso mostré el vector original y el vector según las peticiones del problema 3

A continuación, el código del problema 3:

```
A=[]
n=int(input("Ingrese el tamaño del vector:"))

for i in range(0,n,1):
    elemento=int(input("Ingresa un número:"))
    A.append(elemento)
print(sep=" ")
print("El vector original es el siguiente",A)
B=A
print(sep="\n")
B.insert(0,B.pop())
print("El vector modificado es el siguiente",B)
```

4. Se tiene un arreglo de 15 filas y 12 columnas. Realice un algoritmo que permita leer el arreglo y que calcule y presente los resultados siguientes:

El menor elemento del arreglo; la suma de los elementos de las cinco primeras filas del arreglo; y el total de elementos negativos en las columnas de la quinta a la nueve.

Nota:

- Para poder realizar el arreglo de 15 filas y 12 columnas, se utilizó la función `np.random.randint(-100,100, size=(15, 12))` que crea el arreglo y al mismo tiempo la rellena con valores aleatorios definiendo el rango de estos valores por el usuario en este caso elegí un rango de -100 a 100, de esta manera podemos omitir el relleno de la matriz que en cuestión sería muy tedioso teclear 180 elementos(números).

```

Spyder
File Edit Search Source Run Debug Consoles Projects Tools View Help
F:\Examen 2P-EYAD
untitled38.py x E1_4.py x
4
5 @author: CrIva
6 """
7 #Se tiene un arreglo de 15 filas y 12 columnas. Realice un algoritmo que permita leer el arreglo y que
8 #calcule y presente los resultados siguientes:
9 # (a) El menor elemento del arreglo; (b) la suma de los elementos de las cinco primeras filas del arreglo;
10
11 # y (c) el total de elementos negativos en las columnas de la quinta a la nueve.
12 import numpy as np
13 sumaFilas=[]
14 Count=0
15
16 Matriz = np.random.randint(-100,100, size=(15, 12))
17 #La funcion np.random.randint(Numero_inicial,numero_final,tamaño=(fila,columna,) se utilizo para crear
18 #una matriz especifica rellena con numeros dentro de un rango
19 print(Matriz)
20 # (a) El menor elemento del arreglo
21 min_valor=np.amin(Matriz)#La funcion np.amin() se utiliza para obtener el el elemnto menor de la matriz
22 print("El menor elemento del arreglo:",min_valor), print(sep="\n")
23 # (b) la suma de los elementos de las cinco primeras filas del arreglo;
24 for i in range(0,5,1):
25     fila=sum(Matriz[i])
26     sumaFilas.append(fila)
27
28 print("El siguiente vector muestra la suma de cada fila: ",sumaFilas)
29 print("la suma de la primera fila es:",sumaFilas[0])
30 print("la suma de la segunda fila es:",sumaFilas[1])
31 print("la suma de la tercera fila es:",sumaFilas[2])
32 print("la suma de la cuarta fila es:",sumaFilas[3])
33 print("la suma de la quinta fila es:",sumaFilas[4])
34 print("La suma total de los elementos de las 5 primeras filas es:",sum(sumaFilas)), print(sep="\n")
35 # (c) el total de elementos negativos en las columnas de la quinta a la nueve.
36
37 for i in range(4,9,1):
38     Columnas=Matriz[:, i]
39     for j in range(0,15,1):
40         if Columnas[j]< 0:
41             Count=Count+1
42
43 print("El total de elementos negativos en las columnas 5 a 9 es: ", Count)

```

Captura de pantalla del código correspondiente al problema 4, con algunas especificaciones

```

In [245]: runfile('F:/Examen 2P-EYAD/EJ_4.py', wdir='F:/Examen 2P-EYAD')
[[ 89  11 -55 -58 -2  -6  81  10 -82  23 -72 -52]
 [ -7  58  23 -30 -24 -18 -85  32  81 -23  83  97]
 [-76 -81 -23 -50  21 -43 -35 -76  87  56 -25  22]
 [-74  11  62  10  94  81  40 -16 -34  -2  20 -24]
 [ 30 -16  91  5  29 -75  50 -27  18 -62  38 -60]
 [ 11  84 -26  44 -69  96  19 -52  23 -29  57 -43]
 [-90  24 -76  77  -5 -95  -2  51 -68 -53  72  91]
 [ 16  30  49  27 -26 -95  66 -30  38 -25 -42 -80]
 [-12 -93 -27  18  8 -57 -54  83  76  72  -7  77]
 [ 74 -47 -89 -62  45 -45  26  3 -54  59 -28 -32]
 [ 41  -3  44  14  52  96 -59 -16 -1  27  43 -81]
 [ 80 -67  59  -4 -85  53 -83  51 -47 -65  80 -59]
 [-46  -6  77  84  27  3 -40  44  84 -67  44  54]
 [-94  75 -62 -96 -37 -13  44  18  -3  80 -16  27]
 [-84  23  32  38  16  92 -37  41 -61 -50 -62 -81]]
El menor elemento del arreglo: -96

El siguiente vector muestra la suma de cada fila: [-113, 187, -223, 168, 21]
la suma de la primera fila es: -113
la suma de la segunda fila es: 187
la suma de la tercera fila es: -223
la suma de la cuarta fila es: 168
la suma de la quinta fila es: 21
La suma total de los elementos de las 5 primeras filas es: 40

El total de elementos negativos en las columnas 5 a 9 es: 38

```

Ventana de resultado, en este caso mostré el arreglo de 15 filas y 12 columnas, así mismo mostré el menor el elemento del arreglo, la suma de los elementos de las primeros 5 filas y el total de elementos negativos en las columnas 5 a 9

A continuación, el código del problema 4:

```
import numpy as np
sumaFilas=[]
Count=0

Matriz = np.random.randint(-100,100, size=(15, 12))
print(Matriz)

min_valor=np.amin(Matriz)
print("El menor elemento del arreglo:",min_valor,), print(sep="\n")
for i in range(0,5,1):
    Fila=sum(Matriz[i])
    sumaFilas.append(Fila)

print("El siguiente vector muestra la suma de cada fila: ",sumaFilas)
print("la suma de la primera fila es:",sumaFilas[0])
print("la suma de la segunda fila es:",sumaFilas[1])
print("la suma de la tercera fila es:",sumaFilas[2])
print("la suma de la cuarta fila es:",sumaFilas[3])
print("la suma de la quinta fila es:",sumaFilas[4])
print("La suma total de los elementos de las 5 primeras filas es:",sum(sumaFilas)), print(sep="\n")

for i in range (4,9,1):
    Columnas=Matriz[:, i]
    for j in range(0,15,1):
        if Columnas[j]< 0:
            Count=Count+1

print("El total de elementos negativos en las columnas 5 a 9 es: ", Count)
```

5. Realice un algoritmo que lea dos vectores de cien elementos y que calcule la suma de éstos guardando su resultado en otro vector, el cual se debe presentar en forma impresa

```

Spyder
File Edit Search Source Run Debug Consoles Projects Tools View Help

F:\Examen 2P-EYAD\EJ_5.py

untitled40.py X EJ_5.py X

1  # -*- coding: utf-8 -*-
2  """
3  Created on Sun May  8 11:47:48 2022
4
5  @author: CrIva
6  """
7  #Realice un algoritmo que lea dos vectores de cien elementos y
8  #que calcule la suma de éstos guardando su resultado en otro
9  #vector, el cual se debe presentar en forma impresa
10
11
12  # Nombre_vector=[] para crear un vector vacio,se utiliza para despues ingresarle datos que pidamos
13  A=[]
14  B=[]
15  C=[]
16  n=101#101 ya que range se detiene un dato antes del especificado es decir en el numero 100
17  n1=100
18  # Este ciclo sirve para que podamos ingresar los elementos del vector A y B  segun el tamaño
19  # que hallamos elegido, el ciclo empieza en 1 a n(tamaño del vector), con un paso a 1
20  for i in range(1,n,1): #se cuenta desde uno para omitir el 0
21      A.append(i)
22      B.append(i)
23
24  # Para poder sumar los elementos respectivos del Vector A y el B utilizamos el vector C
25  # el cual sigue vacio , utilizamos C.append para ingresar los resultados de la suma con
26  # (A[i]+B[i]), en este caso el ciclo realiza la suma de los elementos desde 0 hasta n
27  for i in range(0,n1,1):#en este caso se busca la suma de cada elemento,por lo tanto solamente designamos desde
28  #donde se empieza hasta donde termina, lo que el rango es de 0 a 99 lo es igual a 100 elemntos yas que el 0
29  # tambien se cuenta como posicion de un elemento
30      C.append( A[i] + B[i] )
31
32  print("El vector A es:",A,end = "\n\n")
33  print("El vector B es:",B,end = "\n\n")
34  print("El vector C es la suma de los vectores A y B es:",C,end = "\n\n")
35

```

Captura de pantalla del código correspondiente al problema 5, con algunas especificaciones

```

In [250]: runfile('F:/Examen 2P-EYAD/EJ_5.py', wdir='F:/Examen 2P-EYAD')
El vector A es: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,
19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38,
39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58,
59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78,
79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98,
99, 100]

El vector B es: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,
19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38,
39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58,
59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78,
79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98,
99, 100]

El vector C es la suma de los vectores A y B es: [2, 4, 6, 8, 10, 12, 14, 16, 18,
20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58,
60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98,
100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130,
132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162,
164, 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192, 194,
196, 198, 200]

```

Ventana de resultado, en este caso mostré dos vectores A y B la cual tienen 100 elementos cada uno a como se pidió en el problema, se muestra un tercer vector C que muestra la suma del vector A y B de sus respectivos elementos.

A continuación, el código del problema 5:

```
A=[]
B=[]
C=[]
n=101
n1=100
for i in range(1,n,1):
    A.append(i)
    B.append(i)

for i in range(0,n1,1):
    C.append( A[i] + B[i] )

print("El vector A es:",A,end = "\n\n")
print("El vector B es:",B,end = "\n\n")
print("El vector C es la suma de los vectores A y B es:",C,end = "\n")
```