

Examen 2: Ejercicios de Python.

Al momento de responder las preguntas, cerciorense de que les imprima el resultado.

1.- Dando lista $A = [3, 2, 7, 4, 12, 9, 6, 1]$ desarrolla los siguientes 3 ejercicios.

- Imprime la sumatoria de todos los elementos.
- Ordenalos de mayor a menor.
- Haz una nueva lista pero solo con sus elementos pares.

```
 #(Puntos 5)  
 # Imprime la sumatoria de todos los elementos en a = [3, 2, 7, 4, 12, 9, 6, 1].  
A = [3, 2, 7, 4, 12, 9, 6, 1]  
suma=sum(A)  
print(suma)
```

44

```
 #(Puntos 10)  
 # Ordenalos de mayor a menor  
A.sort()  
print(A)
```

[1, 2, 3, 4, 6, 7, 9, 12]

```
 #(Puntos 10)  
 # Haz una nueva lista pero solo con sus elementos pares  
B=[]  
for v in A:  
    if (v % 2==0):  
        B.append(v)  
print(B)
```

[2, 4, 6, 12]

2.- Dale valor a dos variables y realiza 5 funciones que regresen como resultado diferentes operaciones aritmeticas (suma, resta, etc...).

```
 #(Puntos 15)
```

```

#Asumimos que la posicion de los numeros son sus posiciones en las operaciones
def suma(a,b):
    return (a+b)
def resta(a,b):
    return (a-b)
def division(a,b):
    if(b==0):
        return ("Operacion Invalida") #Podriamos regresar un None, pero es mejor dar retroa
    else:
        return(a/b)
def multiplicacion(a,b):
    return(a*b)
def modulo(a,b):
    return (a%b)

a=10
b=0
c=2
print("numeros:")
print(a,b,c)
print("suma: a y c")
print(suma(a,c))
print("resta: a y c")
print(resta(a,c))
print("division: a y b, a y c")
print(division(a,b))
print(division(a,c))
print("multiplicacion: a y c")
print(multiplicacion(a,c))
print("modulo: a y c")
print(modulo(a,c))

numeros:
10 0 2
suma: a y c
12
resta: a y c
8
division: a y b, a y c
Operacion Invalida
5.0
multiplicacion: a y c
20
modulo: a y c
0

```

3.- Dada la lista A = [3, 2, 7, 4, 12, 9, 6, 1] y lista B = [13, 16, 21, 34, 18, 17, 11, 10] crea una lista nueva que contenga solo los datos impares de ambas listas.

```
 #(Puntos 15)
 #Cambiando A y B
A = [3, 2, 7, 4, 12, 9, 6, 1]
B = [13, 16, 21, 34, 18, 17, 11, 10]
C=[]
A.extend(B)
for v in (A):
    if (v%2==1):
        C.append(v)
print(C)
 #print(A)
```

[3, 7, 9, 1, 13, 21, 17, 11]
 [3, 2, 7, 4, 12, 9, 6, 1, 13, 16, 21, 34, 18, 17, 11, 10]

```
 #Sin cambiar A y B
A = [3, 2, 7, 4, 12, 9, 6, 1]
B = [13, 16, 21, 34, 18, 17, 11, 10]
C=[]

def Impares(A,C):
    for v in A:
        if (v%2==1):
            C.append(v)

Impares(A,C)
Impares(B,C)
print(C)
```

[3, 7, 9, 1, 13, 21, 17, 11]

4.- Dadas las variables W = 11, X = 17, Y = 32, Z = 65, crea una función que clasifique por grupos de edad (niño, adolescente, adulto, adulto mayor. (Utiliza el elif)

```
 #(Puntos 15)
 #Usando las variables como los limites de edad
def rango_edad(edad):
    W = 11
    X = 17
```

```

Y = 32
Z = 65
if (edad<=W):
    print("Tu edad es:",end=" " )
    print(edad)
    print("Eres niño")
elif (edad<=X):
    print("Tu edad es:",end=" " )
    print(edad)
    print("Eres adolescente")
elif (edad<=Y):
    print("Tu edad es:",end=" " )
    print(edad)
    print("Eres adulto joven")
elif (edad<Z):
    print("Tu edad es:",end=" " )
    print(edad)
    print("Eres adulto")
elif (edad>=Z):
    print("Tu edad es:",end=" " )
    print(edad)
    print("Eres adulto mayor")

edades=[7, 12, 32, 33,72]
for v in edades:
    rango_edad(v)

```

```

Tu edad es: 7
Eres niño
Tu edad es: 12
Eres adolescente
Tu edad es: 32
Eres adulto joven
Tu edad es: 33
Eres adulto
Tu edad es: 72
Eres adulto mayor

```

#Usando las variables como las que debemos ordenar, y valores de rangos de edad de wikipedia
#<https://es.wikipedia.org/wiki/Ni%C3%B1o>

```

edades=[]
W = 11
X = 17
Y = 32

```

```

Z = 65
edades.append(W)
edades.append(X)
edades.append(Y)
edades.append(Z)

def rango_edad(edad):

    if (edad<=13):
        print("Tu edad es:",end=" " )
        print(edad)
        print("Eres niño")
    elif (edad<=18):
        print("Tu edad es:",end=" " )
        print(edad)
        print("Eres adolescente")
    elif (edad<=40):
        print("Tu edad es:",end=" " )
        print(edad)
        print("Eres adulto joven")
    elif (edad<65):
        print("Tu edad es:",end=" " )
        print(edad)
        print("Eres adulto mayor")
    elif (edad>=65):
        print("Tu edad es:",end=" " )
        print(edad)
        print("Eres adulto mayor")

for v in edades:
    rango_edad(v)

Tu edad es: 11
Eres niño
Tu edad es: 17
Eres adolescente
Tu edad es: 32
Eres adulto joven
Tu edad es: 65
Eres adulto mayor

```

5.- Utilizando cualquiera de las condicionales (for, if, while...) imprime los siguientes patrones tal y como aparecen en el comentario, respetando los espacios y los saltos de renglon.

```
 #(Puntos 10)
#1
#2 2
#3 3 3
#4 4 4 4
#5 5 5 5 5
```

```
def repetir(n,c=1):
    if c <= n:
        m=str(c)
        m=m+' '
        print(m * c)
        c=c+1
        repetir(n,c)
```

```
repetir(5)
```

```
1
2 2
3 3 3
4 4 4 4
5 5 5 5 5
```

```
 #(Puntos 15)
#5 4 3 2 1
#4 3 2 1
#3 2 1
#2 1
#1
```

```
def decreciente(n):
    A=[]
    for i in range(1,n+1):
        A.append(i)
    A.sort(reverse=True)
    #print(A)
    while(len(A)>0):
        for i in A:
            if(i==1):
                print(i)
            else:
                print(i, end = " ")
```

```
A.pop(0)
```

```
decreciente(5)
```

```
5 4 3 2 1
4 3 2 1
3 2 1
2 1
1
```

6.- Con la lista $A = [3, 2, 7, 4, 12, 9, 6, 1]$ y el uso de condicionales, imprime una nueva lista con cada uno de los elementos de la lista A, al cuadrado.

```
#(Puntos 10)
#Cambiando A y usando condicionales
A = [3, 2, 7, 4, 12, 9, 6, 1]
B= []
```

```
def definir_bandera(A):
    if(len(A)==0):
        return False
    else:
        return True
```

```
bandera=definir_bandera
while(bandera(A)):
    B.append(A[0]*A[0])
    A.pop(0)
    bandera=definir_bandera
print(B)
```

```
[9, 4, 49, 16, 144, 81, 36, 1]
```

```
#Sin cambiar A
A = [3, 2, 7, 4, 12, 9, 6, 1]
B= []
for v in A:
    B.append(v*v)
print(B)
```

```
#Sin cambiar A y usando condicionales
C= []
i=0
```

```

if i<len(A):
    while(i<len(A)):
        C.append(A[i]*A[i])
        i=i+1
else:
    print("No hay elementos en A")
print(C)

```

```

[9, 4, 49, 16, 144, 81, 36, 1]
[9, 4, 49, 16, 144, 81, 36, 1]

```

(Extra puntos 5.)

Conteste los tres puntos para obtener la totalidad de los puntos. *Mencione al menos 3 palabras reservadas para el lenguaje de programación de Python.

def return for if else*Describa la diferencia principal entre las listas y las tuplas en Python.

las listas son objeto que podemos alterar luego de su creacion mientras que las tuplas no ejemplo: (si descomentamos lo comentado veremos que marca error)

Si es posible agregar tuplas a las tuplas a travez de “+” (concatenacion de tuplas) pero no podemos eliminar u ordernar una vez esten dentro. Entre otras cosas podremos notar que las tuplas no tienen funciones como .sort , .sorted o .append mientras que las listas si

```

A=[1,2]
print(A)
A.append(3)
A.append(4)
print(A)
T=(1,2)
#T.append(3)
#T=T+(3)
print(T)
T=T+(3,4)
print(T)
#T=T-(3,4)
#print(T)

```

```

[1, 2]
[1, 2, 3, 4]
(1, 2)
(1, 2, 3, 4)

```


*Describa para que sirve el condicional ELIF.

Elif nos ayuda a agregar aun mas casos a un condicional if, es decir nos sirve para lo mismo que una estructura switch en c/c++. Dado que acepta mas rango de condiciones que un switch es mas facil amplio su rango de uso.

if (condicion): #si: esto elif (condicion 2): #sino, si: esto elif (condicion 3: #sino, si: esto . . . elif(condicion 'n'): # sino, si: esto else: #sino: esto