

Nombre:

Cantidad de horas de estudio personal:
Esta semana
Semana pasada
Semana ante-pasada

Problema 1 (20%)

Rutée MANUALMENTE (sin usar el computador) el código de la derecha, e indique el valor que van tomando todas las variables. Escriba los valores usando una tabla como la siguiente:

Table with 4 columns: Nombre variable1, Nombre variable1, Nombre variable1, ...

En el encabezado de la tabla escriba el nombre de la variable, y hacia abajo escriba los valores que adquiere la variable a medida que van cambiando de valor. Escriba un valor por casillero, incluyendo los valores dentro de la lista. Use la siguiente página para escribir su tabla con valores.

```
lista = []

for i in range(1, 7, 3):
    lista.append(i)

i = 2
while i < 6:
    lista.append(i)
    i = i + 2

print(lista)
q = len(lista)
m = 1
c = 0

for a in range(q):
    for b in range(2, q):
        if lista[a] % lista[b] == 0:
            m = m * -1
            c = c + 1
        else:
            c = c - 1

c = c * m
```

Problema2 (35%)

Una fábrica de autos produce tres modelos: sedán, camioneta y económico. Cada auto necesita para su producción una cantidad de unidades de materiales, personal y transporte. La cantidad de unidades son las siguientes:

Table with 4 columns: Model, Materiales, Personal, Transporte

Por otro lado, cada unidad de material tiene un costo, dado en la siguiente tabla:

Table with 2 columns: Material, Costo en \$

Construya un programa que pregunte la cantidad construida de sedanes, camionetas y económicos, y escriba el costo total de construcción.

Ejemplo1:

Ingrese cantidades construidas:
Sedanes: 1
Camionetas: 1
Económicos: 1
Costo total: 459.0

Ejemplo2:

Ingrese cantidades construidas:
Sedanes: 7
Camionetas: 8
Económicos: 12
Costo total: 3981.0

Problema3 (45%)

Debido a la cantidad de sismos detectados en el último mes, se hace necesario crear un sistema que permita analizar los datos recopilados por los sismógrafos. Estos datos incluyen la cantidad de sismos de cierto grado detectados, y el momento del día de la detección (M:Mañana, T:Tarde, N:Noche)

Construya un programa que realice los siguientes pasos:

1. Pregunte el grado, la cantidad y el momento del día en que se realizó la detección. Use un grado 0 como terminador de ciclo. Vea el ejemplo. En el primer grupo de datos, el sismógrafo detectó que hubo 2 sismos de grado 2 durante la tarde.
2. Si se trata de ingresar un valor para un grado y un momento del día que ya tiene datos, se debe indicar que dicha combinación ya fue ingresada previamente.
3. Una vez ingresados todos los datos, el programa debe escribir por pantalla lo siguiente:
 - a. El resumen de los datos, por grado. Para cada grado de sismo, la cantidad de sismos, y el porcentaje que ocurrió en la mañana/tarde/noche.
 - b. El resumen por momento del día. Se debe mostrar el total de sismos, y la cantidad de sismos durante la mañana/tarde/noche. También se debe mostrar el porcentaje ocurrido durante la mañana/tarde/noche.
 - c. El detalle de los sismos considerados fuertes (o sea, con grado 7 o superior). Debe escribir el total, y el porcentaje de dichos sismos ocurridos en la mañana/tarde/noche respecto al total de sismos fuertes.
 - d. El/los grados con más sismos registrados.

Observaciones:

1. La cantidad de grados a considerar es 10 (desde el grado 1 al grado 10)
2. Considere que siempre puede existir más de un elemento que cumpla el criterio de búsqueda (vea el ejemplo).
3. En la siguiente página dibuje las estructuras de datos que utilizará para almacenar los datos y resolver el problema, especificando todas las matrices, vectores que va a usar, y qué guardará en cada una.

Observaciones: El archivo .py de cada problema se debe subir a Educa, en la sección "Trabajos". La plataforma Educa se cierra automáticamente. No hay entregas pasada la hora de término indicada en Educa. La hora oficial es la que dice Educa, **NO** la que dice el computador. Para asegurarme que leyó completamente estas instrucciones, agregue al código un comentario que contenga su nombre completo dentro de las primeras cinco líneas. Si el código no contiene el comentario, se descontará un porcentaje de la nota final. Una prueba respondida correctamente en un 60% corresponde a una nota 4.0.

Compromiso de honestidad

Mediante esta firma me comprometo a ser honesto al realizar esta evaluación, y a que entregaré el resultado de mi trabajo personal. (Al que no firma, no se le revisará la prueba)

```
Grado del sismo: 2
Cantidad de sismos: 2
Momento del día: T
Grado del sismo: 3
Cantidad de sismos: 3
Momento del día: N
Grado del sismo: 4
Cantidad de sismos: 2
Momento del día: M
Grado del sismo: 7
Cantidad de sismos: 2
Momento del día: T
Grado del sismo: 7
Cantidad de sismos: 1
Momento del día: N
Grado del sismo: 8
Cantidad de sismos: 1
Momento del día: M
Grado del sismo: 2
Cantidad de sismos: 2
Momento del día: T
Ya hay un dato para el grado 2 y T
Grado del sismo: 0

Resumen por grado
-----
Grado 1
    Total 0.0: 0/0/0
Grado 2
    Total 2.0: 0.0/100.0/0.0
Grado 3
    Total 3.0: 0.0/0.0/100.0
Grado 4
    Total 2.0: 100.0/0.0/0.0
Grado 5
    Total 0.0: 0/0/0
Grado 6
    Total 0.0: 0/0/0
Grado 7
    Total 3.0: 0.0/66.6667/33.3333
Grado 8
    Total 1.0: 100.0/0.0/0.0
Grado 9
    Total 0.0: 0/0/0
Grado 10
    Total 0.0: 0/0/0

Resumen por momento del dia
-----
    Total 11.0 : 3.0/4.0/4.0
    Porcentajes: 27.27/36.36/36.36

Detalle de sismos fuertes
-----
    Total: 4.0
    Porcentajes: 25.0/50.0/25.0

Grados con más sismos (3.0)
-----
    Grado 3
    Grado 7
```

Firma

Problema 1. Escriba acá la tabla con los valores de las variables

Problema 3. Dibuje acá la estructura de datos que va a utilizar, indicando qué almacenará en cada lista y matriz:

Prueba 2 Programación – 2016-1
Duración: 3 horas

01 Julio 2016

Solucion1:

Prueba 2 Programación – 2016-1
Duración: 3 horas

01 Julio 2016

Solución2:

```
# -*- coding: utf-8 -*-
```

```
import numpy as np
```

```
A = [  
    [7, 8, 5],  
    [10, 9, 7],  
    [3, 4, 2]  
]
```

```
print("Ingrese cantidades construidas:")
```

```
B = [  
    float(input("Sedanes: ")),  
    float(input("Camionetas: ")),  
    float(input("Económicos: "))  
]
```

```
costos = [ 6, 12, 3 ]
```

```
X = np.dot(A, B)
```

```
#print(X)
```

```
T = np.dot(X, costos)
```

```
print("Costo total: " + str(T))
```

Prueba 2 Programación – 2016-1
Duración: 3 horas

01 Julio 2016

Solución3:

```
# -*- coding: utf-8 -*-
```

```
import numpy as np
```

```
datos = np.zeros([10, 3])
momentos = np.zeros([3])
resumen = np.zeros([3, 3])
```

```
grado = int(input("Grado del sismo: "))
```

```
while grado != 0:
```

```
    cantidad = int(input("Cantidad de sismos: "))
    momento = input("Momento del día: ")
```

```
    if momento == "M":
```

```
        fila = 0
```

```
    elif momento == "T":
```

```
        fila = 1
```

```
    else:
```

```
        fila = 2
```

```
    if datos[grado-1][fila] != 0:
```

```
        print("Ya hay un dato para el grado " + str(grado) + " y " + momento)
```

```
    else:
```

```
        datos[grado-1][fila] = cantidad
```

```
        if grado >= 1 and grado <= 4:
```

```
            fr = 0
```

```
        elif grado >= 5 and grado <= 6:
```

```
            fr = 1
```

```
        elif grado >= 7:
```

```
            fr = 2
```

```
        resumen[fr][fila] = cantidad
```

```
        momentos[fila] = momentos[fila] + cantidad
```

```
    grado = int(input("Grado del sismo: "))
```

```
print("Resumen por grado")
```

```
print("-----")
```

```
maxCantidad = -1
```

```
for g in range(10):
```

```
    s = datos[g][0] + datos[g][1] + datos[g][2]
```

```
    if s > maxCantidad:
```

```
        maxCantidad = s
```

```
print("Grado " + str(g + 1))
```

```
p1 = 0
```

```
p2 = 0
```

```
p3 = 0
```

```
if s > 0:
```

```
    p1 = 100 * datos[g][0] / s
```

```
    p2 = 100 * datos[g][1] / s
```

```
    p3 = 100 * datos[g][2] / s
```

```
print("\tTotal " + str(s) + " : " + str(p1) + "/" + str(p2) + "/" + str(p3))
```

```
print("Resumen por momento del día")
```

```
print("-----")
```

```
s = momentos[0] + momentos[1] + momentos[2]
```

```
print("\tTotal " + str(s) + " : " + str(momentos[0]) + "/" + str(momentos[1]) + "/" + str(momentos[2]))
```

```
p1 = 0
```

```
p2 = 0
```

```
p3 = 0
if s > 0:
    p1 = 100 * momentos[0] / s
    p2 = 100 * momentos[1] / s
    p3 = 100 * momentos[2] / s

print("\tPorcentajes: " + str(p1) + "/" + str(p2) + "/" + str(p3))

print("Detalle de sismos fuertes")
print("-----")

s = resumen[2][0] + resumen[2][1] + resumen[2][2]

print("\tTotal: " + str(s))

p1 = 0
p2 = 0
p3 = 0
if s > 0:
    p1 = 100 * resumen[2][0] / s
    p2 = 100 * resumen[2][1] / s
    p3 = 100 * resumen[2][2] / s

print("\tPorcentajes: " + str(p1) + "/" + str(p2) + "/" + str(p3))

print("Advertencias del mes")
print("-----")

for g in range(10 - 1):
    if (datos[g][2] > 0 and datos[g+1][0] > 0):
        print("\tPrecaución grado " + str(g + 1) + "/" + str(g + 2))

print("Grados con más sismos (" + str(maxCantidad) + ")")
print("-----")

for g in range(10):
    s = datos[g][0] + datos[g][1] + datos[g][2]
    if s == maxCantidad:
        print("\tGrado " + str(g + 1))
```