

### CERTAMEN 3 – PROGRAMACIÓN PYTHON

TIEMPO (120 MIN)

1. (50 %)

Pythonic SA es una importante productora y distribuidora de productos orgánicos, la cual cuenta con un producto estrella y sus 3 vendedores, Larry, Curly y Moe. La gerencia de esta empresa, con el afán de obtener indicadores de gestión sobre la venta de su producto estrella, le ha pedido a usted escribir un programa que permita realizar las siguientes operaciones con la información de las ventas de cada vendedor:

Operación	Significado
1	Ingresar la venta, esta opción solicita el nombre del vendedor y la cantidad vendida.
2	Venta mayor, esta opción muestra la venta mayor realizada y quién la realizó.
3	Peor vendedor, esta opción muestra el vendedor que menos productos ha vendido a la fecha, considerando todas sus ventas ingresadas.
0	Salir, esta opción debe mostrar el total vendido por cada vendedor y Salir / terminar el programa, de lo contrario vuelve a solicitar una operación.

Validar que las operaciones ingresadas correspondan a las señaladas arriba, de no existir señalar que *"La operación no existe"*. Si dos o tres vendedores empatan en el número mínimo o máximo de ventas, indique cualquiera de ellos en las opciones respectivas. Además asuma que todos los datos ingresados son correctos.

A continuación se presenta un **ejemplo** de cómo debiera funcionar el programa:

```
1.- Ingresar ventas
2.- Venta mayor
3.- Peor vendedor
0.- Salir
Operacion? 1
Nombre vendedor: larry
Cantidad vendida: 15
Operacion? 1
Nombre vendedor: moe
Cantidad vendida: 35
Operacion? 1
Nombre vendedor: curly
Cantidad vendida: 24
Operacion? 1
Nombre vendedor: larry
Cantidad vendida: 22
Operacion? 3
El vendedor que menos a vendido es curly
Operacion? 1
Nombre vendedor: curly
Cantidad vendida: 10
Operacion? 2
La mayor venta realizada fue de 35 y la realizo moe
Operacion? 0
moe vendio: 35 productos
larry vendio: 37 productos
curly vendio: 34 productos
```

2. (50%)

Para un proyecto de iluminación con celdas solares se dispone de una batería, la cual se carga de día y se descarga de noche. La autonomía de una batería se define como el tiempo que demora en descargarse, el cual puede ser calculado mediante la ley de Peukert:

$$t = \frac{H}{\left(\frac{IH}{C}\right)^k}$$

$t$	Autonomía de la batería en [Horas]
$C$	Capacidad de la batería en [Ampere-Hora]
$H$	Base de tiempo definida por el fabricante en [Horas]
$I$	Consumo en [Amperes]
$k$	Constante de Peukert

Se desea saber cuál es el número máximo de ampolletas que se puede conectar a esta batería de modo que puedan permanecer encendidas toda la noche, es decir, que la autonomía sea **mayor o igual a 8 horas**. Se sabe que el consumo  $I$ , el cual es empleado en la ley de Peukert, viene dado por la suma de las potencias de cada ampolleta dividida por el voltaje de la batería, es decir:

$$I = \frac{\text{PotenciaAmpolleta}_1 + \text{PotenciaAmpolleta}_2 + \dots}{\text{VoltajeBateria}}$$

Mientras que la constante de Peukert  $k$  tiene un valor de 1,15

Desarrolle un programa Python que solicite los datos de la batería y luego solicite la potencia de cada ampolleta. El programa debe detenerse cuando no se puedan agregar más ampolletas debido a que la autonomía total sería menor que la autonomía requerida. Luego de solicitar la potencia de cada ampolleta, el programa debe imprimir la autonomía, la cantidad de ampolletas y la potencia total acumulada.

```
Capacidad bateria (AH): 33
Voltaje bateria (volt): 12
Base de tiempo (Horas): 20

Potencia ampolleta 1 (Watt): 12
Autonomia: 33.837 [Horas]. Ampolletas: 1. Potencia Total: 12.0 [Watt]
Potencia ampolleta 2 (Watt): 12
Autonomia: 16.342 [Horas]. Ampolletas: 2. Potencia Total: 24.0 [Watt]
Potencia ampolleta 3 (Watt): 32

Total de Ampolletas: 2
```

#### CONSIDERACIONES:

- VALIDE TODO LOS INGRESOS DE DATOS EN LOS DOS EJERCICIOS
- USE TRY-EXCEPT