

**Universidad Tecnológica de Panamá**  
**Facultad de Ingeniería en Sistemas Computacionales**  
**Asignatura: Desarrollo Lógico Y Algoritmo**  
**Ejercicio Práctico2**

Profesor: Napoleón Ibarra

Valor: 100 puntos

Estudiante: Jafet Loo 4-838-349

Ihammal Medina 4-835-706

**Fecha Inicio: 13/10/2025 --> 4:10 PM**

**Fecha Entrega: 14/10/2025 -->2:25 PM**

**Fecha Cierre (Tope): 20/10/2025 -->4:10 PM**

**Procedimiento:**

- ✓ De manera INDIVIDUAL o en grupos de 2 PERSONAS, realizar la asignación.
- ✓ Investigue, descargar e instalar en una máquina virtual (Usted elige el IDE) una distribución de Windows Server (2016/2019).
- ✓ Se debe entregar al profesor: Documento digital: entrega en la plataforma (TEAM) el y/o los códigos desarrollando los problemas. Sustente su trabajo en el aula de clases.

**Criterios de Evaluación:**

<b>Criterios</b>	<b>Puntos (Mínimo=1, Máximo=5)</b>	<b>Porcentaje</b>
Desarrollo	1-5	70 %
Sustentación	1-5	15 %
Puntualidad	1-5	15 %

**I Parte. Laboratorio. Valor 20 puntos**

1. Elegir un software para virtualización (Virtual Box / VMware), descargar, instalar y configurarlo.
2. Descargar, instalar la distribución de Windows Server (2016/2019) elegida dentro de la máquina virtual.
3. Realizar el proceso de configuración correspondiente al realizar el proceso de la instalación del SO.
4. Configurar una carpeta para hacer respaldos de códigos dentro del SO, verifique su instalación, configuración.
5. Realizar la configuración del Servidor de Datos (que se accesible por Remote Desktop Connection, ANYDESK, FTP).
6. Hacer pruebas de funcionamiento.

**II Parte. Caso de Estudio. Valor 20 Puntos**

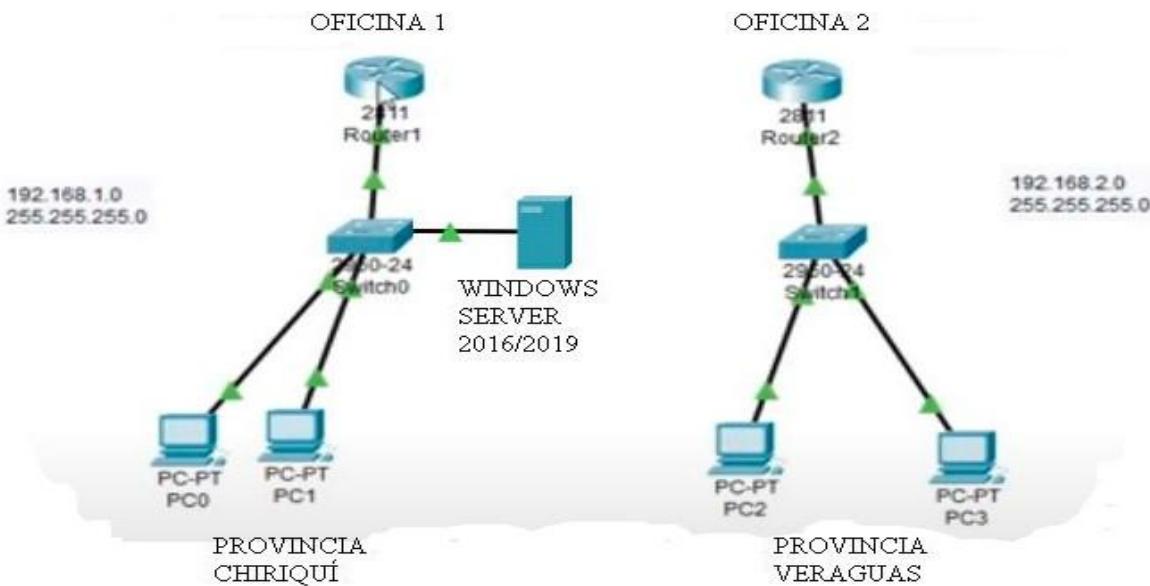


Figura 1. Prototipo de diagrama

**Procedimiento:**

1. Teniendo en cuenta el siguiente plano de oficina (Figura 1), confeccione el esquema de RED LAN, puesto que en ambas oficinas 1 y 2 se va a utilizar el servidor de datos.

### **III Parte. Pseudocódigos, diagrama de flujo. *Valor 10 Puntos***

1. Desarrolle los problemas en pseudocódigo para la parte IV de la asignación.

### **IV Parte. Desarrollo de problemas en Python. *Valor 20 Puntos***

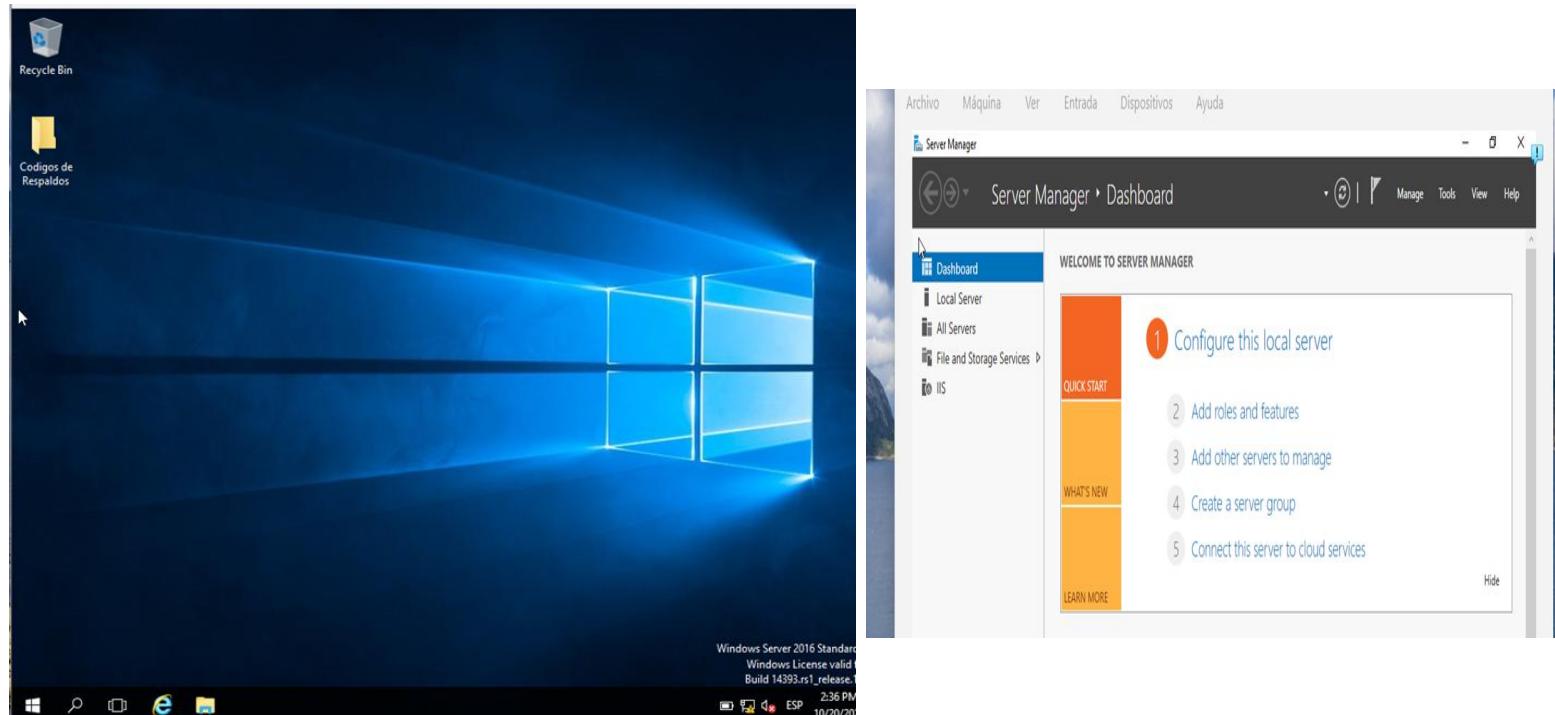
1. Escriba un programa en Python que lea la hora en notación de 24 horas y que imprima en notación de 12; por ejemplo, si la entrada es 13:45, la salida será 1:45 pm. El programa debe solicitar al usuario que introduzca exactamente cinco caracteres para especificar una hora. *Valor 10 puntos*
2. Una compañía de refrescos comercializa tres productos: de cola, de naranja y de limón. Se dese realizar una aplicación que calcule las ventas realizadas de cada producto. Para ello, se leerá la cantidad vendida (máximo 5000000) y el precio en Balboas de cada producto y se mostrará un informe de ventas como el que sigue:

Producto	Ventas	Precio	Total
<hr/>			
Cola	1000000	0.17	170000.00
Naranja	350000	0.20	70000.00
Limon	530000	0.19	100700.00
	TOTAL		340700.00

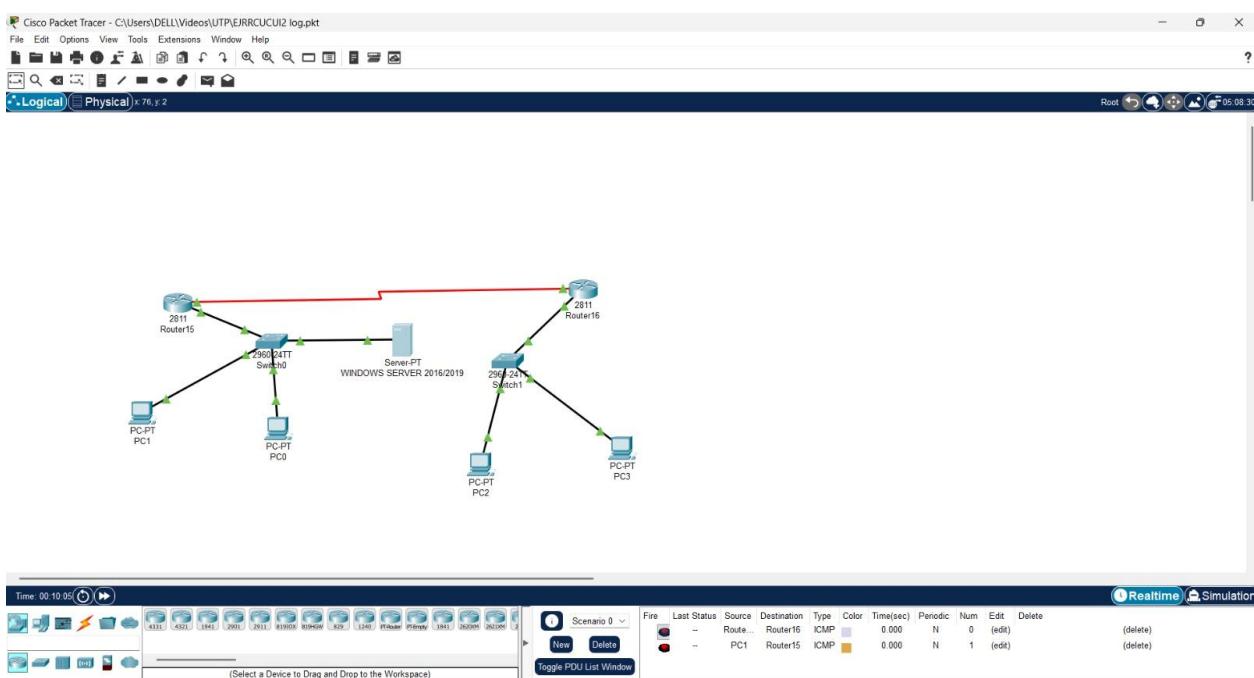
NOTA: Para el desarrollo de la parte IV en Python, utilizar la interfaz Tkinter, tome en cuenta después del punto (decimal) 2 cifras significativas.

**BUENA SUERTE**

### **I Parte. Laboratorio**

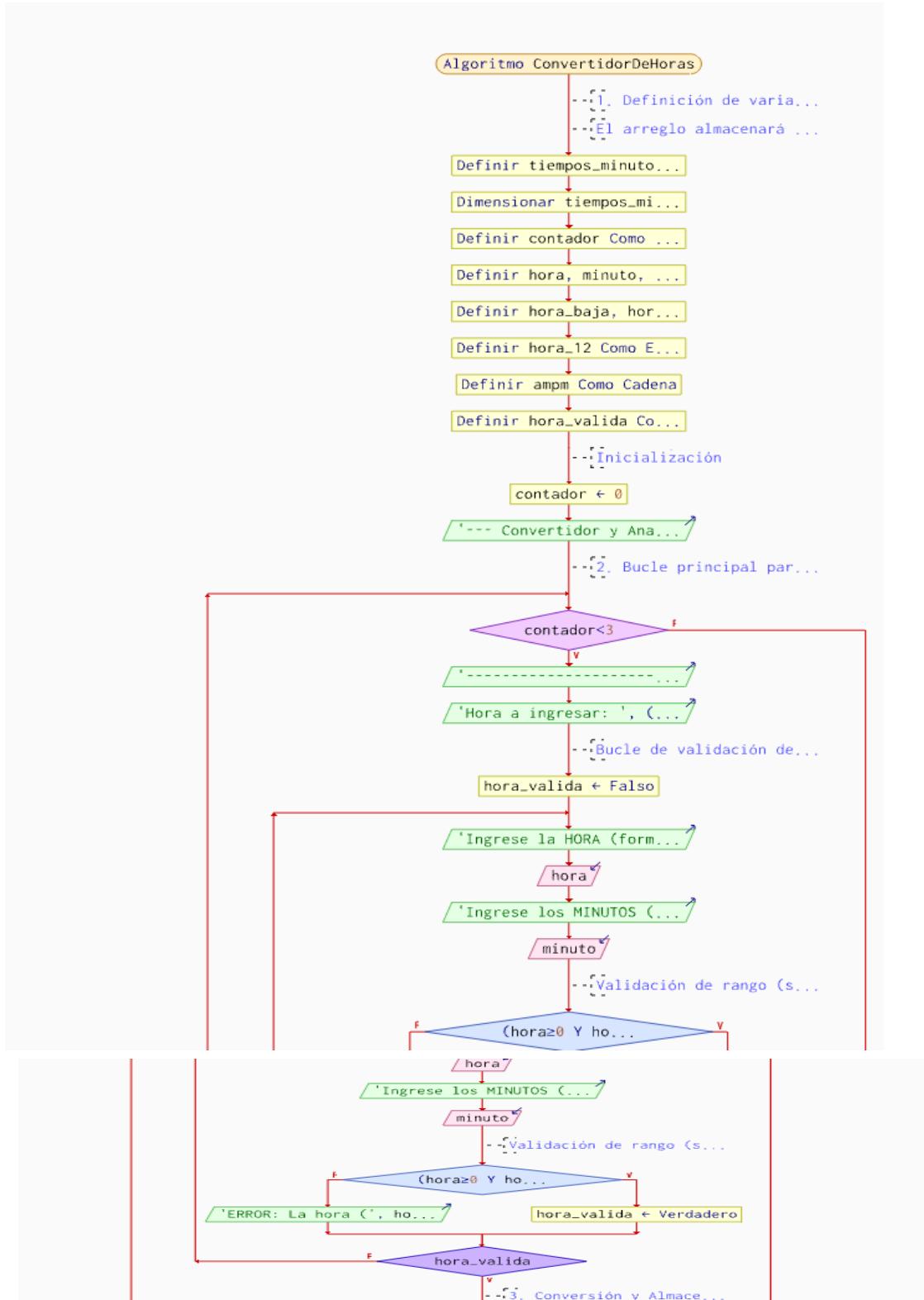


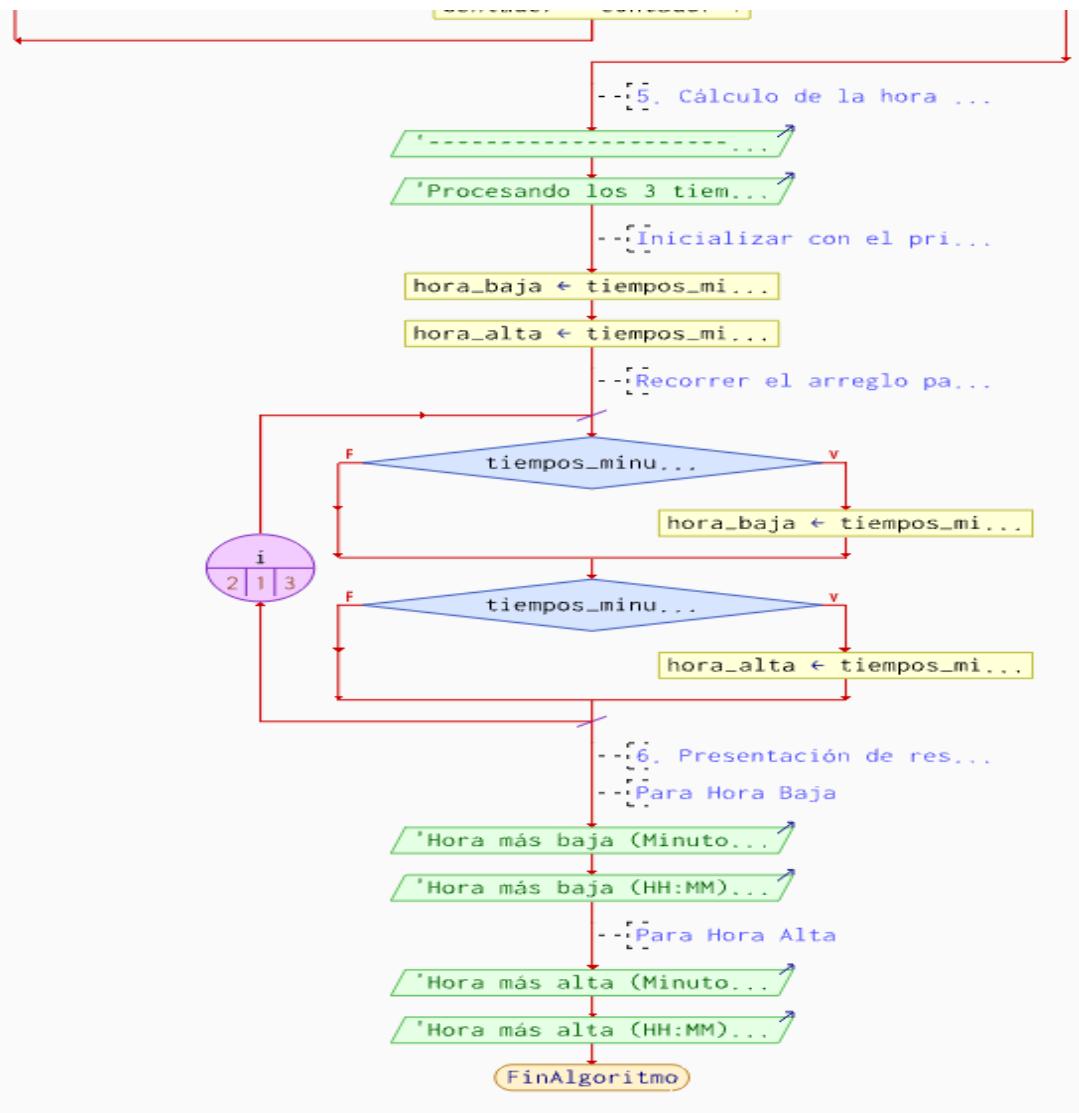
## II Parte. Caso de Estudio.



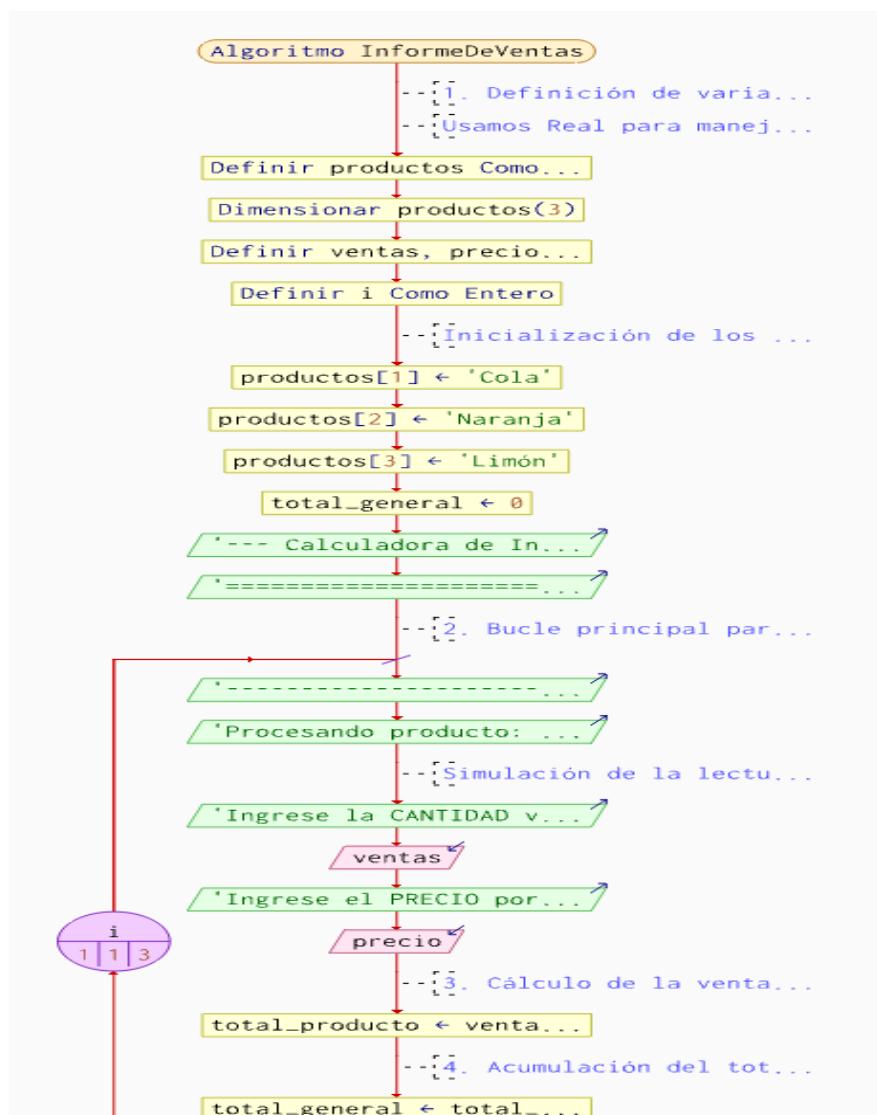
### Parte. Pseudocódigos, diagrama de flujo

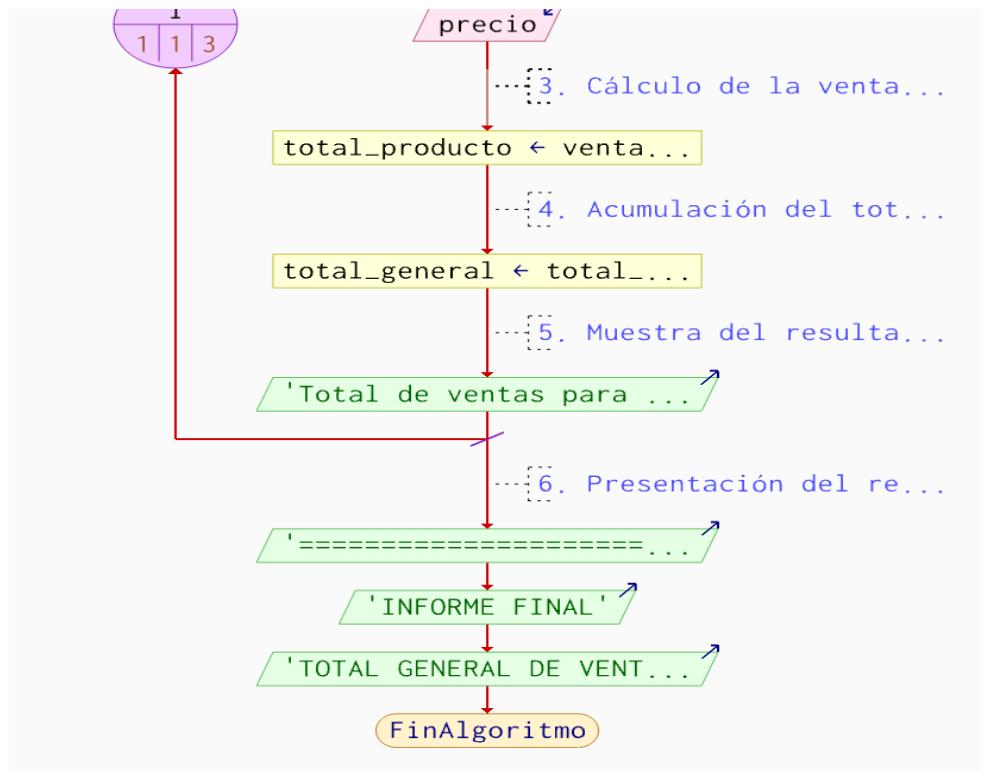
Primer problema ConvertirHoras





### Segundo Problema Ventaproductos





## IV. Parte Lenguaje en Python

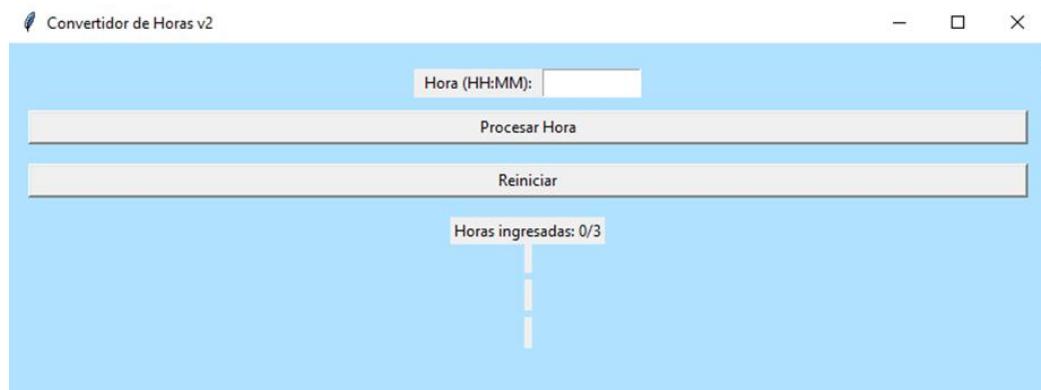
1.

### Primer Problema

```

nueva_pruvea.py Notas de la versión: 1.105.0 LeerHoras.PY ConvertirHoras.pyprimer problema.dslpy horas.py
1 import tkinter as tk
2
3 class TimeApp(tk.Tk):
4     def __init__(self):
5         super().__init__() # Llama al constructor de la clase padre (tk.Tk)
6         self.super(
7             t: Any,
8             obj: Any,
9             /
10         ): ...
11
12         self.super(
13             t: Any,
14             /
15         ): ...
16
17         self.super(): ...
18
19
20     def _crear_widgets(self):
21         """Método privado para crear y posicionar todos los componentes."""
22         # --- Frame de entrada ---
23         frame_entrada = tk.Frame(self, bg="#F0F0F0")
24         tk.Label(frame_entrada, text="Hora (HH:MM):", bg="#F0F0F0").pack(side="left", padx=5)
25         self.campo_texto = tk.Entry(frame_entrada, width=8, font=("Arial", 12))
26         self.campo_texto.pack(side="left")
27         frame_entrada.pack(pady=5)
28
29         # --- Botones ---
30         self.boton_procesar = tk.Button(self, text="Procesar Hora", command=self.procesar_hora)
31         self.boton_procesar.pack(pady=5, fill="x")
32
33         tk.Button(self, text="Reiniciar", command=self.reiniciar).pack(pady=(10, 15), fill="x")
34
35         # --- Etiquetas de resultados ---
36         self.label_resultado = tk.Label(self, text="Horas ingresadas: 0/3", font=("Arial", 12))
37         self.label_resultado.pack(side="bottom", fill="x", pady=10)

```



## Segundo Problema

```
nueva prueba.py Notas de la versión: 1.105.0 LeerHoras.PY ConvertirHoras.pyprimer problema.ds1.py horas.py Ventasdproducto.py 1
Ventasdproducto.py > ...
1 import tkinter as tk
2 from tkinter import ttk, messagebox
3
4 # --- VALORES PREDETERMINADOS Y ESTRUCTURA DE DATOS ---
5 DATOS_INICIALES = {
6     "Cola": ("1000000", "0.17"),
7     "Naranja": ("350000", "0.20"),
8     "Limón": ("530000", "0.19")
9 }
10 COLOR_FONDO = "SeaGreen1"
11 COLOR_TEXTO = "white"
12 COLOR_RESALTADO = "yellow"
13
14 class AppVentas:
15     """Clase que encapsula la lógica del algoritmo y la interfaz."""
16     def __init__(self, master):
17         self.master = master
18         master.title("Informe de Ventas de Refrescos")
19         master.geometry("600x400")
20         master.maxsize(600, 600)
21         master.configure(bg=COLOR_FONDO)
22         master.resizable(True, False)
23
24         # Estilos para Entry y Button
25         style = ttk.Style()
26         style.configure('Entry', font=('Arial', 10))
27         style.configure('Button', font=('Arial', 10, 'bold'))
28
29         # Diccionarios para almacenar variables de Tkinter
30         self.vars_precio = {}
31         self.vars_cantidad = {}
32         self.vars_resultado = {}
33         self.var_total_general = tk.StringVar(value="0.00")
34
```

