

Ejercicio Práctico2

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

Profesor: Napoleón Ibarra

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

Estudiante: Jacob Samudio

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

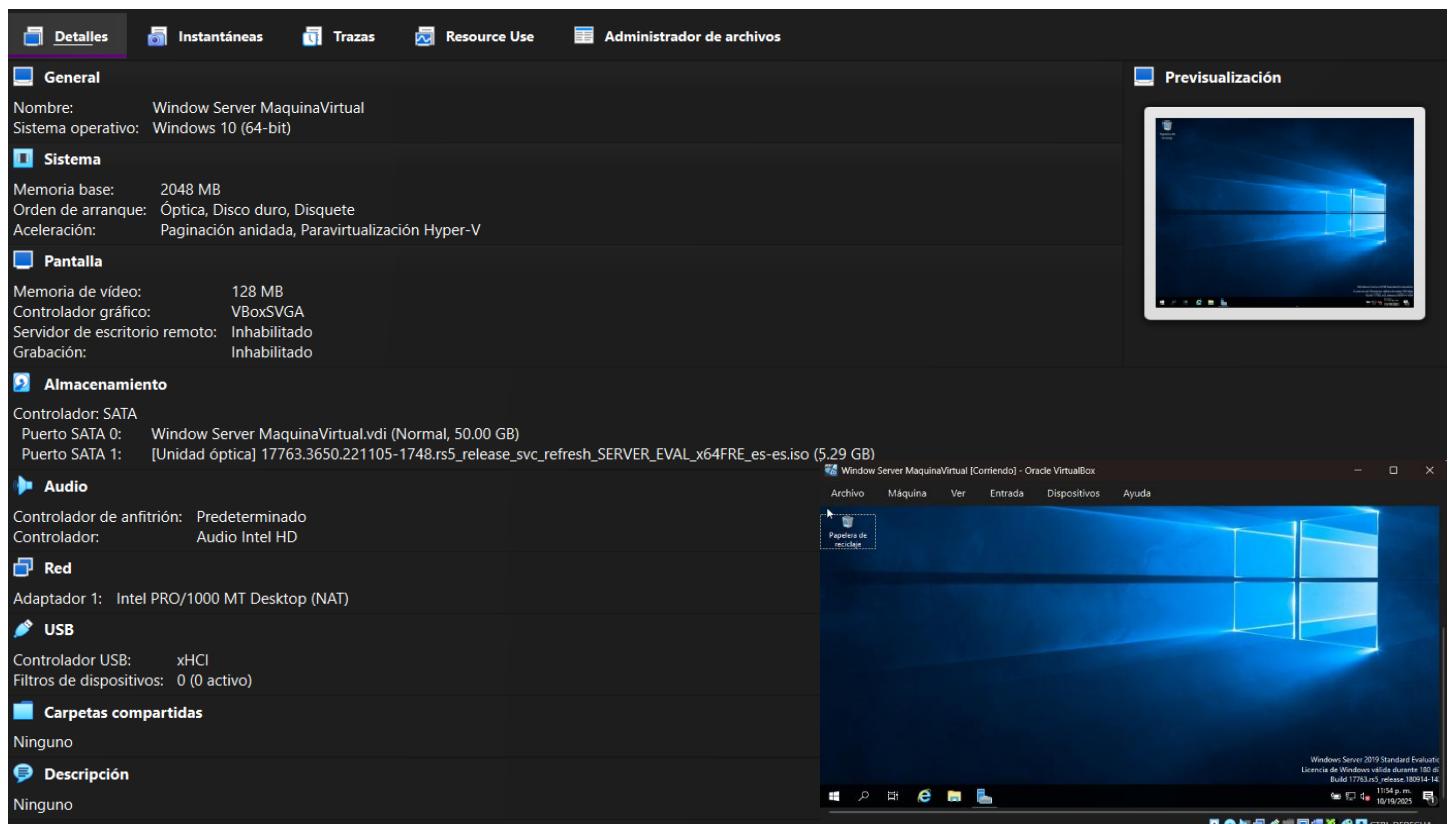
Diógenes Serrano

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.

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.



```
Adaptador de Ethernet Ethernet:
Sufijo DNS específico para la conexión. . . : lan
Dirección IPv6 . . . . . : fd17:625c:f037:2:f746:53bf:4ec5:1675
Vínculo: dirección IPv6 local. . . : fe80::66b8:4e48:d5ac:fb4b%7
Dirección IPv4. . . . . : 10.0.2.15
Máscara de subred . . . . . : 255.255.255.0
Puerta de enlace predeterminada . . . . : fe80::2%7
10.0.2.2
```

Nombre	Fecha de modifica...	Tipo	Tamaño
Acceso rápido			
Escritorio	10/20/2025 12:02 a...	Carpetas de archivos	
Descargas	10/20/2025 12:02 a...	Carpetas de archivos	
Documentos	10/20/2025 12:03 a...	Carpetas de archivos	

PARTE I - LABORATORIO: COMPLETADO

Software de virtualización: VirtualBox instalado y configurado

Windows Server 2019 Standard instalado exitosamente

Configuración de red: IP 10.0.2.15 (asignación automática NAT)

Carpeta de respaldos: C:\Backups creada con subcarpetas Scripts, Database, Logs

Servidor de datos configurado con:

Escritorio Remoto habilitado

AnyDesk operativo para acceso remoto

Servidor FTP instalado (IIS)

Pruebas de funcionamiento exitosas via AnyDesk

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.26100.6899]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\user>ping 10.0.2.15

Haciendo ping a 10.0.2.15 con 32 bytes de datos:
Respueta desde 10.0.2.15: bytes=32 tiempo=9ms TTL=58
Respueta desde 10.0.2.15: bytes=32 tiempo=83ms TTL=58
Respueta desde 10.0.2.15: bytes=32 tiempo=108ms TTL=58
Respueta desde 10.0.2.15: bytes=32 tiempo=8ms TTL=58

Estadísticas de ping para 10.0.2.15:
Paquetes: enviados = 4, recibidos = 4, perdidos = 0
(0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 8ms, Máximo = 108ms, Media = 52ms

C:\Users\user>
```

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

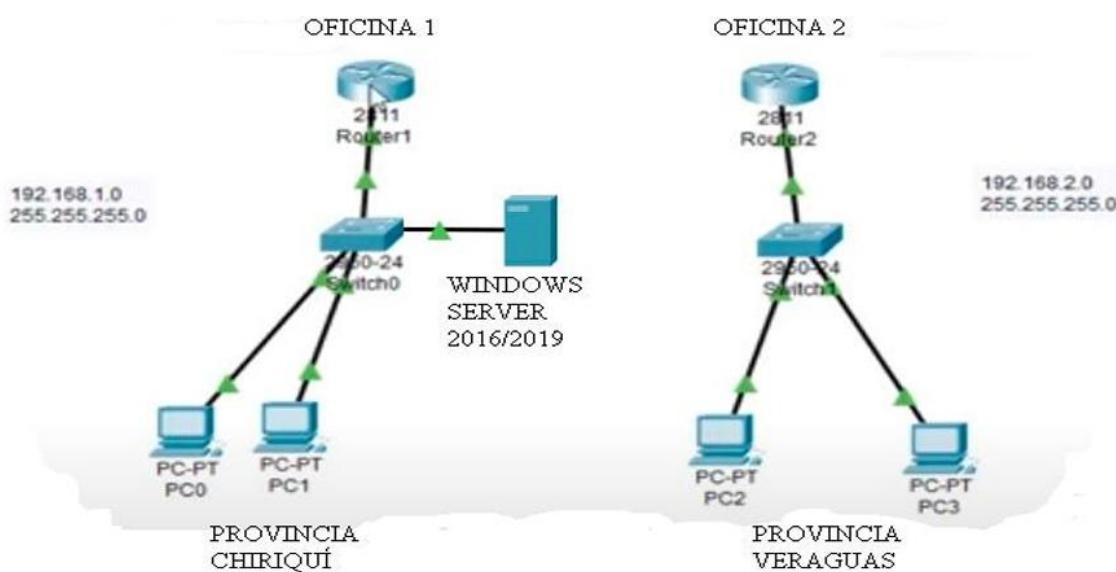
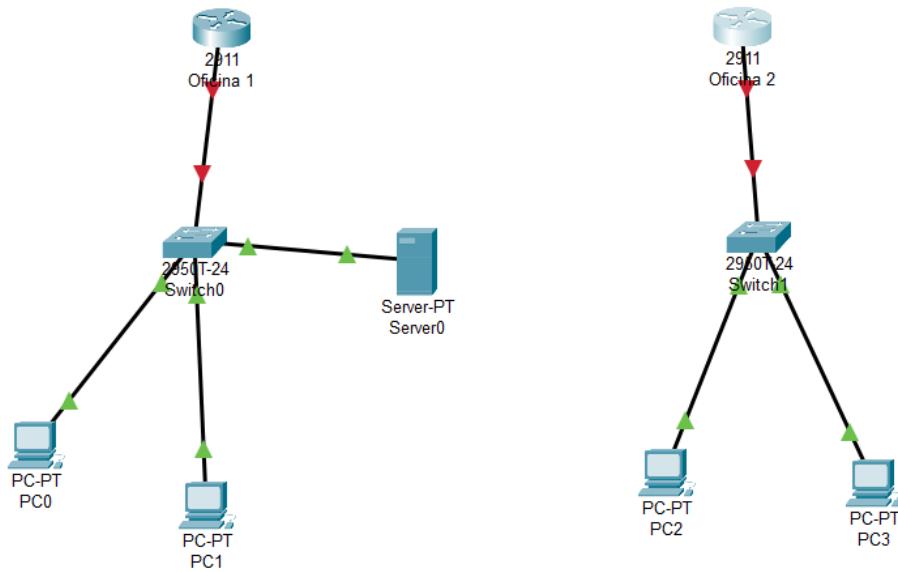


Figura 1. Prototipo de diagrama

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.



Command Prompt

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.20.3

Pinging 192.168.20.3 with 32 bytes of data:

Request timed out.
Reply from 192.168.20.3: bytes=32 time=1ms TTL=126
Reply from 192.168.20.3: bytes=32 time=1ms TTL=126
Reply from 192.168.20.3: bytes=32 time=1ms TTL=126

Ping statistics for 192.168.20.3:
  Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
  Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 1ms, Average = 1ms

C:\>
```

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

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

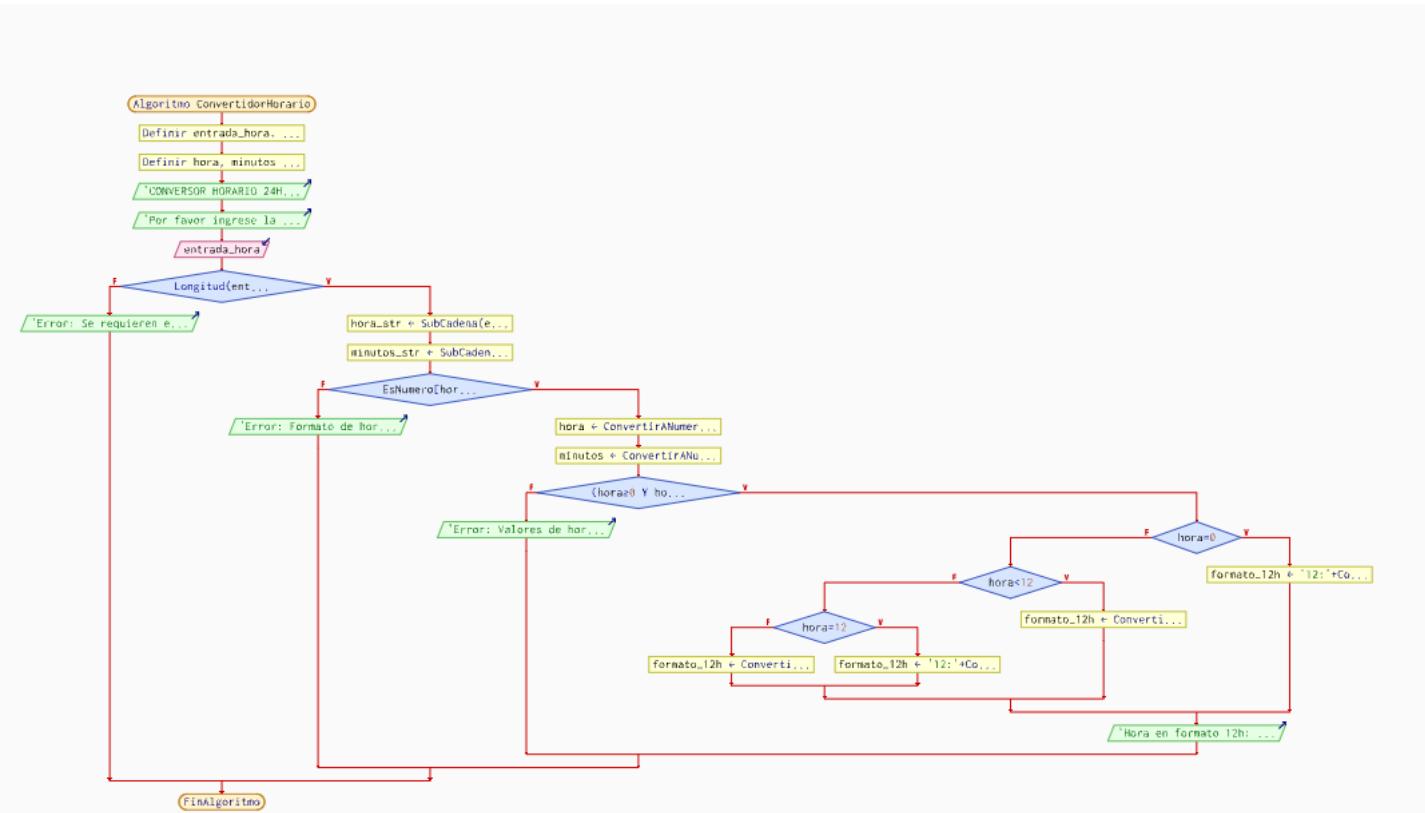
Desarrollo

Problema 1.

```

1 Algoritmo ConvertidorHorario
2   Definir entrada_hora, hora_str, minutos_str, formato_12h Como Cadena
3   Definir hora, minutos Como Entero
4
5   Escribir "CONVERSOR HORARIO 24H ? 12H"
6   Escribir "Por favor ingrese la hora (formato HH:MM):"
7   Leer entrada_hora
8
9   Si Longitud(entrada_hora) = 5 Entonces
10    hora_str ← SubCadena(entrada_hora, 1, 2)
11    minutos_str ← SubCadena(entrada_hora, 4, 5)
12
13 +  Si EsNumero(hora_str) Y EsNumero(minutos_str) Entonces
14    hora ← ConvertirANumerico(hora_str)
15    minutos ← ConvertirANumerico(minutos_str)
16
17    Si (hora ≥ 0 Y hora ≤ 23) Y (minutos ≥ 0 Y minutos ≤ 59) Entonces
18      Si hora = 0 Entonces
19        formato_12h ← "12:" + ConvertirATexto(minutos) + " a.m."
20      Sino
21        Si hora < 12 Entonces
22          formato_12h ← ConvertirATexto(hora) + ":" + ConvertirATexto(minutos) + " a.m."
23        Sino
24          Si hora = 12 Entonces
25            formato_12h ← "12:" + ConvertirATexto(minutos) + " p.m."
26          Sino
27            formato_12h ← ConvertirATexto(hora - 12) + ":" + ConvertirATexto(minutos) + " p.m."
28          FinSi
29        FinSi
30      FinSi
31
32      Escribir "Hora en formato 12h: ", formato_12h
33    Sino
34      Escribir "Error: Valores de hora o minutos inválidos."
35    FinSi
36  Sino
37    Escribir "Error: Formato de hora inválido. Ingrese números."
38  FinSi
39  Sino
40    Escribir "Error: Se requieren exactamente 5 caracteres (ejemplo: 09:45)."
41  FinSi
42 FinAlgoritmo
43

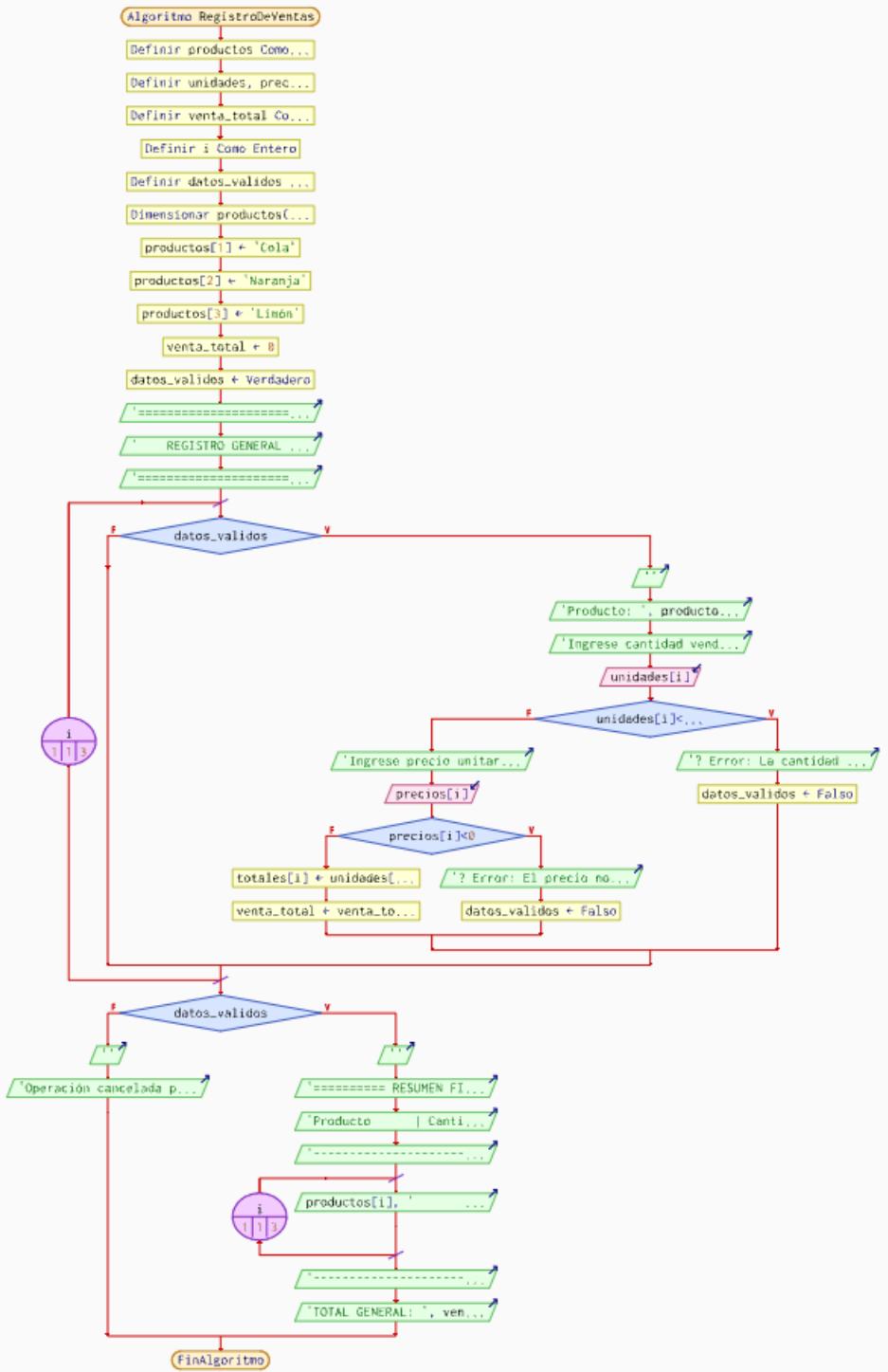
```



```

1 Algoritmo RegistroDeVentas
2   Definir productos Como Cadena
3   Definir unidades, precios, totales Como Real
4   Definir venta_total Como Real
5   Definir i Como Entero
6   Definir datos_validos Como Logico
7
8   Dimension productos[3], unidades[3], precios[3], totales[3]
9
10  productos[1] <- "Cola"
11  productos[2] <- "Naranja"
12  productos[3] <- "Limon"
13
14  venta_total <- 0
15  datos_validos <- Verdadero
16
17  Escribir "*****"
18  Escribir " REGISTRO GENERAL DE VENTAS"
19  Escribir "*****"
20
21  Para i <= 1 Hasta 3 Con Paso 1 Hacer
22    Si datos_validos Entonces
23      Escribir ""
24      Escribir "Producto: ", productos[i]
25      Escribir "Ingrese cantidad vendida:"
26      Leer unidades[i]
27
28      Si unidades[i] < 0 O unidades[i] > 5000000 Entonces
29        Escribir "? Error: La cantidad debe estar entre 0 y 5,000,000."
30        datos_validos <- Falso
31
32      Sino
33        Escribir "Ingrese precio unitario:"
34        Leer precios[i]
35
36      Si precios[i] < 0 Entonces
37        Escribir "? Error: El precio no puede ser negativo."
38        datos_validos <- Falso
39      Sino
40        totales[i] <- unidades[i] * precios[i]
41        venta_total <- venta_total + totales[i]
42      FinSi
43    FinSi
44  FinPara
45
46  Si datos_validos Entonces
47    Escribir ""
48    Escribir "===== RESUMEN FINAL ====="
49    Escribir "Producto | Cantidad | Precio | Importe"
50    Escribir "-----"
51
52    Para i <= 1 Hasta 3 Con Paso 1 Hacer
53      Escribir productos[i], "      ", unidades[i], "      ", precios[i], "      ", totales[i]
54  FinPara
55
56  Escribir "-----"
57  Escribir "TOTAL GENERAL: ", venta_total, " Balboas"
58
59  Sino
60    Escribir ""
61    Escribir "Operación cancelada por datos invalidos."
62  FinSi
63 FinAlgoritmo

```



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	1700000.00
Naranja	350000	0.20	700000.00
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.

Desarrollo

PROBLEMA 1

```
# CONVERSION DE HORAS - FORMATO 24H A 12H
print("PROGRAMA DE CONVERSION HORARIA")
hora_input = input("Digite la hora en notacion de 24h (HH:MM): ")

if len(hora_input) == 5:
    try:
        horas_componente = int(hora_input[0:2])
        minutos_componente = hora_input[3:5]
        minutos_valor = int(minutos_componente)

        if 0 <= horas_componente <= 23 and 0 <= minutos_valor <= 59:
            if horas_componente == 0:
                salida = f"12:{minutos_componente} a.m."
            elif horas_componente < 12:
                salida = f"{horas_componente}:{minutos_componente} a.m."
            elif horas_componente == 12:
                salida = f"12:{minutos_componente} p.m."
            else:
                salida = f"{horas_componente-12}:{minutos_componente} p.m."

            print(f"Resultado: {salida}")
        else:
            print("Error: Los valores estan fuera de rango")
    except:
        print("Error: Formato numerico incorrecto")
else:
    print("Error: Se necesitan exactamente 5 caracteres")
```

Problema 2

```
# SISTEMA DE REGISTRO DE VENTAS DE BEBIDAS
print("REGISTRO DE VENTAS - PRODUCTOS BEBIBLES")
print("=====")

productos_lista = ["Refresco Cola", "Refresco Naranja", "Refresco Limón"]
unidades = []
precios_unidad = []
```

```
totales_parciales = []
suma_general = 0
proceso_ok = True

for producto_actual in productos_lista:
    if not proceso_ok:
        break

    print(f"\nRegistrando: {producto_actual}")

    try:
        cantidad = int(input("Número de unidades vendidas: "))
        if cantidad < 0 or cantidad > 5000000:
            print("Cantidad fuera del rango permitido")
            proceso_ok = False
            continue

        precio = float(input("Valor unitario en Balboas: "))
        if precio < 0:
            print("El precio no puede ser negativo")
            proceso_ok = False
            continue

        total_parcial = cantidad * precio

        unidades.append(cantidad)
        precios_unidad.append(precio)
        totales_parciales.append(total_parcial)
        suma_general += total_parcial

    except:
        print("Entrada numérica inválida")
        proceso_ok = False

if proceso_ok and len(unidades) == 3:
    print("\n" + "*55)
    print("REPORTE DETALLADO DE VENTAS")
    print("*55)
    print(f"{'Producto':<18} {'Cantidad':<12} {'Precio':<12} {'Total':<12}")
    print("-*55)

    for i in range(3):
        print(f"{productos_lista[i]:<18} {unidades[i]:<12} B/.{precios_unidad[i]:<10.2f}
B/.{totales_parciales[i]:<10.2f}")

    print("-*55)
    print(f"{'SUMA TOTAL':<42} B/.{suma_general:.2f}")
else:
    print("\nNo se pudo completar el registro correctamente")
```