**Examen Práctico – Programación no numérica I**

Fecha: 28/04/25

Duración: 2 horas académicas

Nombre del Estudiante: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

C.I: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Instrucciones Generales:**

1. Lea atentamente cada ejercicio antes de comenzar a codificar.
2. Cree un archivo .c separado para cada ejercicio. Nombre los archivos como se indica (ej., ejercicio1.c, ejercicio2.c, etc.).
3. Escriba código claro, legible y bien comentado.
4. Compile y pruebe su código frecuentemente para asegurarse de que funciona correctamente.
5. Preste atención a los tipos de datos requeridos y al formato de salida solicitado.
6. Al finalizar, guarde todos sus archivos .c en una carpeta con su nombre y C.I (ej., NombreApellido\_Cedula)
7. Está prohibido el uso de internet, teléfonos móviles o cualquier material no autorizado explícitamente por el profesor. Solo se permite el uso del editor de texto y el compilador C.

**Puntaje Total: 20 puntos**

**Ejercicio 1: Tipos de Datos y Operadores Aritméticos (5 puntos)**

Archivo: **ejercicio1.c**

Descripción:

Escriba un programa en C que realice lo siguiente:

1. Declare tres variables:

* Una de tipo int llamada cantidad.
* Una de tipo float llamada precioUnitario.
* Una de tipo double llamada total.

1. Solicite al usuario que ingrese un valor entero para la cantidad de productos.
2. Solicite al usuario que ingrese un valor decimal (con punto flotante) para el precioUnitario del producto.
3. Calcule el total multiplicando cantidad por precioUnitario.
4. Calcule el Impuesto al Valor Agregado (IVA) sobre el total. Suponga una tasa de IVA del 16% (0.16). Guarde este valor en una variable float llamada iva.
5. Calcule el totalPagar sumando el total y el iva. Guarde este valor en una variable double.
6. Muestre en pantalla los valores de cantidad, precioUnitario, total, iva y totalPagar, formateados adecuadamente (el precioUnitario, total, iva y totalPagar deben mostrarse con 2 decimales).

**Ejercicio 2: Estructuras de Control (switch, for) y Funciones (5 puntos)**

Archivo: **ejercicio2.c**

Descripción:

Escriba un programa en C que:

1. Defina una función llamada imprimirTablaMultiplicar que reciba un parámetro entero numero y no retorne ningún valor (void). Esta función debe:

* Imprimir un encabezado indicando la tabla que se mostrará (ej., "Tabla de multiplicar del 5:").
* Usar un bucle for para iterar del 1 al 10.
* Dentro del bucle, calcular e imprimir cada línea de la tabla de multiplicar para el numero dado (ej., "5 x 1 = 5", "5 x 2 = 10", ..., "5 x 10 = 50").

1. En la función main:

* Declare una variable entera opcion.
* Muestre un menú simple al usuario:

|  |
| --- |
| **--- Menu Tablas de Multiplicar ---**  **1. Tabla del 3**  **2. Tabla del 7**  **3. Tabla del 9**  **Seleccione una opcion:** |

1. Lea la opcion ingresada por el usuario.
2. Utilice una estructura switch basada en la opcion ingresada:

* Si opcion es 1, llame a imprimirTablaMultiplicar(3).
* Si opcion es 2, llame a imprimirTablaMultiplicar(7).
* Si opcion es 3, llame a imprimirTablaMultiplicar(9).

1. Si opcion no es ninguna de las anteriores, muestre un mensaje "Opción no válida."

**Ejercicio 3: Arreglos (5 puntos)**

Archivo: ejercicio3.c

Descripción:

Escriba un programa en C que:

1. Defina una constante TAMANIO con valor 5.
2. Declare un arreglo de enteros llamado numeros de tamaño TAMANIO.
3. Use un bucle for para solicitar al usuario que ingrese TAMANIO números enteros y guárdelos en el arreglo numeros.
4. Use otro bucle for para calcular la suma de todos los elementos almacenados en el arreglo.
5. Muestre en pantalla la suma total calculada.

**Ejercicio 4: Estructuras (struct) (5 puntos)**

Archivo: ejercicio4.c

Descripción:

Escriba un programa en C que:

1. Defina una estructura llamada Producto que contenga los siguientes miembros:

* int codigo;
* char nombre[50]; (arreglo de caracteres para el nombre)
* float precio;

1. En la función main:

* Declare una variable de tipo struct Producto llamada miProducto.
* Solicite al usuario que ingrese el código (entero), nombre (cadena de texto) y precio (flotante) del producto.
* Almacene los datos ingresados en los miembros correspondientes de la variable miProducto.
* Muestre en pantalla los datos almacenados en la estructura, con un formato claro.