



Nombre: _____ fecha:

5/03/2024 Trainer: Alberto Galvis

Grupo: **Dell**

Reglas de la prueba

- **Comunicación:** Está prohibido hablar o comunicarse de cualquier forma con otros estudiantes durante el examen.
- **Integridad Académica:** Cualquier forma de trampa, incluido el plagio, copiado o uso de material no autorizado, resultará en una calificación de cero en el examen y puede llevar a sanciones adicionales según las políticas de RIWI.
- **Material Permitido:** Solo se permite ver material de apoyo como lo son diapositivas o ejercicios realizados en clase, y notas tomadas en el cuaderno.
- **Permanencia en el Aula:** Una vez iniciado el examen, no se permite salir del aula hasta haber entregado el examen y, de preferencia, hasta que haya transcurrido al menos la mitad del tiempo asignado.
- **Entrega:** Una vez finalizada la prueba se debe enviar en una carpeta comprimida: Simulacro1_NombreApellido del coder al correo electrónico alberto.galvis@riwi.io, en la carpeta comprimida deben estar los tres ejercicios, cada uno en un archivo independiente. La hora máxima de entrega es a las 11:55 a.m, las pruebas enviadas después de esta hora se tendrán en cuenta

Ejercicio 1: Gestión de Lista de Compras

Nivel: Alto

Descripción del Problema:

Una familia necesita una aplicación para gestionar su lista de compras semanal. Cada semana, la familia elabora una lista de productos que necesitan comprar en el supermercado. La aplicación debe permitir a los usuarios agregar, modificar, eliminar, listar y buscar productos de manera eficiente.

Requisitos del Problema:

- **Agregar Producto:** Los usuarios deben poder agregar un nuevo producto a la lista de compras. Cada producto debe incluir un nombre descriptivo, una cantidad requerida y, opcionalmente, una nota adicional.
- **Modificar Producto:** Los usuarios deben tener la capacidad de modificar un producto existente en la lista de compras. Esto incluye cambiar la cantidad requerida o la nota adicional asociada al producto.
- **Eliminar Producto:** Los usuarios deben poder eliminar un producto existente de la lista de compras si ya no es necesario.
- **Listar Productos:** Se debe poder mostrar una lista de todos los productos en la lista de compras, incluyendo detalles como el nombre del producto, la cantidad requerida y la nota adicional.
- **Buscar Productos:** Los usuarios deben poder buscar un producto específico en la lista de compras utilizando su nombre o cualquier otro criterio relevante.

Notas:



Interfaz de Usuario Amigable: El programa debe proporcionar una interfaz de usuario intuitiva a través de un **menu** que guíe a los usuarios a través de las diferentes opciones disponibles para administrar las tareas y los proyectos.



Ejercicio 2: Calculadora de Descuentos por Cantidad de Artículos

Nivel: Medio

Descripción del Problema:

Una tienda en línea desea implementar una calculadora de descuentos para sus clientes. El negocio ofrece descuentos basados en la cantidad de artículos que el cliente compra. Se aplican descuentos diferentes según la cantidad de artículos comprados.

Requisitos del Problema:

Entrada de Datos: El programa debe solicitar al usuario que ingrese la cantidad de artículos que desea comprar.

Calcular Descuento: El programa debe calcular el descuento aplicable según la cantidad de artículos comprados, utilizando la siguiente tabla:

- Si el cliente compra menos de 5 artículos, no se aplica ningún descuento.
- Si el cliente compra entre 5 y 10 artículos (inclusive), se aplica un descuento del 5%.
- Si el cliente compra entre 11 y 20 artículos (inclusive), se aplica un descuento del 10%.
- Si el cliente compra más de 20 artículos, se aplica un descuento del 15%.

Mostrar Resultado: El programa debe mostrar el descuento aplicado y el monto total a pagar después de aplicar el descuento correspondiente.

Notas:



Interfaz de Usuario Amigable: El programa debe proporcionar una interfaz de usuario intuitiva a través de un **menu** que guíe a los usuarios a través de las diferentes opciones disponibles para administrar las tareas y los proyectos.



Ejercicio 3: Calculadora de Propina

Nivel: Bajo

Descripción del Problema:

Escribe un programa en Python que ayude a calcular la propina a dejar en un restaurante. El programa debe permitir al usuario ingresar el total de la cuenta y calcular la propina basada en un porcentaje de 12%.

Requisitos del Problema:

Entrada de Datos: El programa debe solicitar al usuario que ingrese el total de la cuenta en el restaurante.

Calcular Propina: El programa debe calcular la propina basada en el total de la cuenta y el porcentaje de propina equivalente al 12%.

Mostrar Resultado: El programa debe mostrar al usuario el monto de la propina calculada.



Criterio	Puntuación 3 (Excelente)	Puntuación 2 (Bueno)	Puntuación 1 (Necesita Mejorar)	Puntuación 0 (no presenta)
Conocimiento del Anunciado	El estudiante muestra entendimiento del enunciado y cumple con todos los requisitos de manera.	El estudiante demuestra comprensión del enunciado y cumple con la mayoría de los requisitos.	El estudiante tiene dificultades para entender el enunciado y no cumple con los requisitos clave.	El estudiante no demuestra entender ningún enunciado
Calidad del Código	El código del estudiante es limpio, bien estructurado y eficiente. Se siguen las mejores prácticas de programación.	El código del estudiante es generalmente claro y funcional, pero podría haber mejoras en la estructura.	El código del estudiante es desorganizado, difícil de leer y contiene errores graves.	La cantidad de código escrito, no es suficiente para determinar su calidad
Cumplimiento de Requisitos Técnicos	El estudiante cumple con todos los requisitos técnicos del ejercicio de manera precisa y completa.	El estudiante cumple con la mayoría de los requisitos técnicos, pero puede haber omisiones o errores menores.	El estudiante no logra cumplir con varios requisitos técnicos clave.	El estudiante no cumple con ningún requisito técnico
Comentarios y Documentación del Código	El estudiante proporciona comentarios detallados y documentación clara en el código. Los comentarios explican de manera efectiva el propósito y la lógica del código.	El estudiante proporciona comentarios adecuados, pero puede haber algunas áreas que carecen de documentación.	El código del estudiante carece de comentarios o documentación significativa.	El código no evidencia ningún comentario o documentación
Sustentación	El estudiante muestra un profundo entendimiento del código y demuestra conocimiento completo sobre la lógica, estructura, funciones y métodos.	El estudiante demuestra comprensión del código, y la lógica, pero no identifica estructuras, funciones o métodos.	El estudiante tiene dificultades para entender el enunciado no identifica la lógica del código	El estudiante no presenta sustentación

