

EXAMEN PARCIAL – UNIDADES 5 Y 6

TRIMESTRE: PRIMERO

Fecha:

CICLO: Desarrollo de Aplicaciones Web.

CURSO: 1º

CALIFICACIÓN:

MÓDULO: Programación

Turno: Mañana

Nombre:

Apellidos:

Instrucciones: Esta prueba tiene como finalidad evaluar los aprendizajes de Programación. Lee atentamente y responde escribiendo el código más adecuado.

Si las instrucciones no se siguen como se especifican el examen no será evaluado

PARTE PRÁCTICA. TIPO A.

- El examen práctico tiene una puntuación máxima de 10 puntos.
- Para superar la parte práctica se requiere alcanzar un mínimo de 5 puntos.

1. **(3 puntos)** Desarrolla un sistema de gestión de carrito de compras para una tienda en línea. El sistema debe permitir a los usuarios agregar productos al carrito, calcular el total a pagar aplicando un descuento global del 10%, y mostrar los detalles de los productos en el carrito.

Productos Disponibles:

Los productos deben pertenecer a una de las siguientes categorías: ELECTRONICA, ROPA, o ALIMENTACION.

Cada producto tiene un nombre, un precio, y una categoría.

Carrito de Compras:

El carrito tiene una capacidad fija de productos.

Los usuarios pueden agregar productos al carrito siempre que no se supere la capacidad.

El sistema debe calcular el precio total con un descuento global del 10% aplicado a cada producto.

Debe mostrarse el nombre y el precio con descuento de cada producto, así como el total final del carrito.

Operaciones Permitidas:

Agregar un nuevo producto al carrito.

Mostrar los productos en el carrito junto con el total a pagar.

Clases Requeridas:

- CategoriaProducto: Una enumeración que define las categorías de los productos (ELECTRONICA, ROPA, ALIMENTACION).
- Producto: Clase que representa un producto, con los atributos nombre, precio, y categoria. Incluye un método para aplicar un descuento global del 10%.
- Carrito: Clase que maneja una colección de productos, permite agregar productos hasta un límite definido, y calcula el total a pagar con el descuento aplicado.
- Test: Clase principal que simula la creación de productos, su adición a un carrito, y la impresión de los detalles del carrito y el total final.

Resultado de ejecución:

Laptop - Precio final: 1080.0

Camiseta - Precio final: 18.0

Manzanas - Precio final: 2.25

Total: 1100.25

2. (3 puntos) Crea un sistema para gestionar proyectos y tareas dentro de una empresa. Los proyectos tienen un estado (por ejemplo, EN_PROCESO, FINALIZADO, PAUSADO) y cada tarea asociada a un proyecto tiene una prioridad (ALTA, MEDIA, BAJA). Los proyectos tienen un presupuesto y las tareas tienen un costo asociado. La clase Proyecto tiene un atributo estático para el presupuesto total de todos los proyectos. El sistema debe ser capaz de calcular el presupuesto total de los proyectos activos.

- Requisitos:
 - Crea un tipo enumerado EstadoProyecto con los valores EN_PROCESO, FINALIZADO, PAUSADO.
 - Crea un tipo enumerado PrioridadTarea con los valores ALTA, MEDIA, BAJA.
 - Crea una clase Tarea que tenga los atributos nombre, prioridad y costo.
 - Crea una clase Proyecto que tenga los atributos nombre, estado, presupuesto, y un array de Tarea.
 - La clase Proyecto debe calcular el costo total de las tareas en curso y sumarlo al presupuesto.
 - La clase Empresa debe calcular el presupuesto total de los proyectos activos (con estado EN_PROCESO).

Resultado de ejecución:

Proyecto: Proyecto A

Estado: EN_PROCESO

Presupuesto final: 12600,00€

Proyecto: Proyecto B

Estado: FINALIZADO

Presupuesto final: 11700,00€

Proyecto: Proyecto C

Estado: EN_PROCESO

Presupuesto final: 13900,00€

Presupuesto total de proyectos en proceso: 26500,00€

3. (4 puntos) Crea un sistema para gestionar el alquiler de vehículos en una empresa. Los vehículos tienen una categoría (ELECTRICO, SUV, SEDAN), y cada categoría tiene un descuento específico. Además, cada cliente tiene un historial de alquileres y se le aplica un descuento adicional dependiendo de la cantidad de alquileres previos. Los vehículos deben ser gestionados mediante un array, y se debe calcular el precio final de cada alquiler considerando los descuentos por categoría y por historial del cliente.

- Requisitos:
 - Crea un tipo enumerado `CategoriaVehiculo` con los valores `ELECTRICO`, `SUV`, `SEDAN`.
 - Crea una clase `Vehiculo` con los atributos `matricula`, `categoria`, `precioDiario` y un método `calcularPrecioAlquiler`.
 - Crea una clase `Cliente` con los atributos `nombre`, `numeroAlquileres`, `vehiculosAlquilados[]` y un método `calcularDescuentoCliente`.
 - Los vehículos deben ser gestionados en un array dentro de una clase `Alquiler`.
 - Los descuentos son:
 - Eléctrico: 15%
 - SUV: 10%
 - SEDAN: No tiene descuento
 - Crea un método estático en `Alquiler` que calcule el precio final de un alquiler para un cliente dado, tomando en cuenta los descuentos por categoría del vehículo y por historial del cliente.

Resultado de ejecución:

Cliente: Juan Pérez

Vehículo: ABC123 - ELECTRICO

Precio final por 3 días: 127,50€

Cliente: Ana López

Vehículo: LMN789 - SEDAN

Precio final por 2 días: 120,00€