

PORTAFOLIO EJERCICIO DE CÁTEDRA N 2

FECHA DE ENTREGA AL ESTUDIANTE: 20 de mayo.

FECHA DE ENTREGA AL DOCENTE: 5 de junio, hasta antes de la medianoche

CONDICIONES DEL PORTAFOLIO:

El ejercicio #2 del Portafolio de Aprendizaje del curso EIF200 Fundamentos de Informática tiene el propósito de que el estudiante ponga en práctica la resolución de problemas de programación utilizando clases de objetos y arreglos unidimensionales en C++, IDE Zinjai, según lo establecido por la cátedra.

El ejercicio se debe trabajar en parejas, sólo un estudiante deberá subir la solución en la actividad correspondiente en el aula virtual. El nombre del archivo comprimido con la solución debe incluir el número de ejercicio del portafolio, el número de grupo y el nombre de ambos estudiantes. Por ejemplo: Port2G10RosaRyJuanA.rar. Cualquier copia o plagio entre grupos o estudiantes se calificará con 0, como lo establece el artículo 24 del Reglamento General sobre los Procesos de Enseñanza y Aprendizaje de la Universidad Nacional.

Para este ejercicio se debe adjuntar la documentación tal como se especifica en el documento **Normas para la presentación de proyectos**, disponible en el Aula Virtual. Además, se debe hacer una defensa del proyecto ante el profesor en el plazo que se defina.

Enunciado del problema:

Usted ha sido contratado por una empresa comercial de desarrollo de software para que programe una aplicación que ayude en la administración y toma de decisiones de un supermercado. El supermercado tiene productos de todo tipo y se desea que el sistema logre mantener control de todos sus productos, cantidades vendidas, existencias, entre otros aspectos.



Los datos más importantes de los productos son los que se describen a continuación. **código:** una hilera de caracteres que identifica de forma única a cada producto; **nombre:** nombre del producto; **precio base:** precio que paga el supermercado por el producto, también llamado precio de costo;

porcentaje de ganancia: valor entre 0 y 1 que representa el porcentaje de ganancia que obtiene el supermercado por la venta del producto y permite calcular su precio de venta, por ejemplo, si porcentaje de ganancia es 0.15, el supermercado obtiene una ganancia del 15% sobre el precio base del producto; **cantidad vendida:** cantidad de unidades del producto que se han vendido; **existencia:** cantidad de unidades del producto que están disponibles en el supermercado, **peso unitario:** peso de cada unidad del producto en kilogramos y; **existencia mínima:** cantidad mínima de unidades del producto que debe mantenerse en el supermercado.

Se le ha pedido a usted que ayude al supermercado a organizar la información sobre los productos que vende. La cantidad máxima de productos que puede manejar el supermercado es de 200, pero aún no se ha llegado a ese límite. Se debe tener registro de los productos que maneja el supermercado, mostrar los datos de los productos, agregar nuevos productos, eliminar productos que ya no se van a vender, etc. Para llevar el control de todos los productos que existen en el supermercado es necesario mantenerlos en una estructura de datos que permita realizar las acciones que se solicitan.

Se requiere que usted defina las clases necesarias para resolver el problema adecuadamente. Debe escribir los métodos básicos de cada clase: los constructores, los accesores, los mutadores y el método toString. Además, debe escribir los métodos particulares que se describen a continuación.

- 1) `ingresaProducto();` **(4 pts.):** Ingresa un nuevo producto al surtido del supermercado.
- 2) `eliminarProductoPorSu();` **(4 pts.):** Elimina un producto del surtido del supermercado por su código.
- 3) `Producto obtenerProducDeMayorValor();` **(4 pts.):** devuelve el producto de mayor precio de venta.
- 4) `Producto obtenerProducConMayorExistencia();` **(4 pts.):** devuelve el producto que tiene mayor existencia.
- 5) `ordenarProductPorCodigo();` **(4 pts.):** ordena los productos ascendentemente de acuerdo con el código.
- 6) `imprimirTodosLosProductos();` **(4 pts.):** muestra los siguientes datos de todos los productos del supermercado: código, nombre, porcentaje de ganancia y precio de venta.
- 7) `cantDeProductosBajosDeExistencia();` **(4 pts.):** Devuelve la cantidad de productos que tienen una existencia por debajo de su existencia mínima.
- 8) `imprimirProductosBajosDeExistencia();` **(4 pts.):** muestra la lista de los productos

que tienen una existencia por debajo de su existencia mínima. Se debe mostrar solamente: el código, el nombre, la existencia y la existencia mínima.

9) `cuantosKgsSeHanVendidoEnGeneral();` **(4 pts.):** devuelve la suma de kilogramos que corresponde a las cantidades vendidas de todos los productos.

10) `cuantosKgsSeHanVendidoDelProducto();` **(4 pts.):** devuelve la cantidad de kilogramos que corresponde con la cantidad vendida de un producto.

11) `imprimProductosQueSonMasCostososQue();` **(4 pts.):** muestra un listado de los productos cuyo precio de venta es mayor que el de un producto en particular que se recibe como parámetro. Solamente se debe mostrar para cada producto: el código, el nombre y su precio de venta.

12) `cuantasUnidadesSeHanVendidoDelProducto();` **(4 pts.):** devuelve la cantidad de unidades vendidas de un producto particular.

13) `costoDelInventario();` **(4 pts.):** devuelve el costo total de los productos en existencia en el supermercado, es decir, la suma de los precios de costo ponderados por la cantidad en existencia de cada producto.

14) `costoDeLasVentas();` **(4 pts.):** devuelve el costo total de los productos que se ha vendido, es decir, la suma de los precios de costo ponderados por la cantidad vendida de cada producto.

15) `valorDeLasVentas();` **(4 pts.):** devuelve el valor total de las ventas del supermercado, es decir, la suma de los precios de venta ponderados por la cantidad vendida de cada producto.

16) `ganaciaDelSuperMercadoPorVentas();` **(4 pts.):** devuelve la ganancia total del supermercado por las ventas realizadas.

17) `graficoDeVentasDeLos15ProductsMasVendidos();` **(6 pts.):** Muestra un gráfico de barras horizontales, dibujadas con asteriscos (*), que representan la cantidad vendida de los 5 productos más vendidos en el supermercado. En las ordenadas (eje Y) se mostrarán los nombres de los productos y en las abscisas (eje X) la representación de la cantidad vendida del producto. Por ejemplo, solamente para fines ilustrativos, vea el siguiente gráfico.



18) `graficoDeVentasDeLos15ProductsMenosVendidos();` **(5 pts.):** Muestra un gráfico análogo al descrito en el ítem anterior para los 5 productos menos vendidos del supermercado.

19) `promedioDeLosPreciosDeVenta ();` **(5 pts.):** devuelve el promedio de los precios de venta de todos los productos que se han vendido. Es decir, tome los precios de venta de los productos vendidos y obtener el promedio de todos ellos.

Menú (Interfaz de Usuario). (5 pts.)

Se requiere que usted implemente un menú de opciones que devuelva un valor que representa la opción escogida por el usuario, entre las opciones del 1 al 20, para facilitar el trabajo de implementación en el main. Acordarse de validar la entrada en el momento de seleccionar la opción. Se sugiere que utilice un menú como el siguiente:

-----MENU-----

- 1- Ingresar un producto.
- 2- Eliminar un producto.
- 3- Desplegar el producto de mayor precio.
- 4- Desplegar el producto con mayor existencia.
- 5- Ordenar los productos por código.
- 6- Desplegar todos los productos.
- 7- Desplegar la cantidad de productos bajos de existencia.
- 8- Desplegar los productos bajos de existencia.
- 9- Desplegar el total de kilogramos de todos los productos vendidos.
- 10- Desplegar el total de Kilogramos vendidos de un producto.
- 11- Desplegar la lista de productos que son más costosos que un producto dado.
- 12- Desplegar la cantidad de unidades vendidas de un producto en particular.
- 13- Desplegar el Costo del Inventario.
- 14- Desplegar el Costo de las Ventas.
- 15- Desplegar el Valor de las Ventas.
- 16- Desplegar la ganancia total del supermercado por concepto de ventas.
- 17- Desplegar el gráfico de las ventas de los 5 productos más vendidos.
- 18- Desplegar el gráfico de las ventas de los 5 productos menos vendidos.
- 19- Desplegar el promedio de precios de venta de los productos vendidos con ganancia.

-20- SALIR

El main(). (5 pts.)

Usted como programador deberá escribir el main() para crear un objeto SuperContenedorV y usarlo con todos los métodos solicitados, de acuerdo con la opción del menú que seleccione el usuario. Se requiere que el menú esté dentro de una estructura iterativa de tal forma que siempre esté disponible para que el usuario seleccione su próxima opción.

Rúbrica

Concepto	Puntaje
Implementación de la Clase SuperContenedorV y sus métodos, incluidos los gráficos	80 pts.
Implementación del main(), del menú y la usabilidad del programa.	10 pts.
Documentación y defensa del proyecto, de acuerdo con las normas establecidas.	10 pts.

Fuente de la imagen que se muestra al inicio de este enunciado:

<https://www.istockphoto.com/es/vector/tienda-supermercado-estantes-anaqueles-con-productos-ilustraci%C3%B3n-de-vector-gm657794780-119958143>