



Facultad de Ingeniería y Arquitectura
Ingeniería en Sistemas de Información

Lógica y Algoritmos

Primer Semestre 2024

PRESENTACIÓN DEL CASO DE ESTUDIO FINAL

Estudiantes:

Cristopher Amaru Rodriguez Arauz

Diego David Fonseca Moody

Harald Antonio Corea Salas

Gabriel Alejandro Rodriguez Ruiz

Managua, mayo de 2024

Introducción.

Dentro del mundo actual, la eficiencia y la precisión son esenciales para el éxito de cualquier tipo de negocio. Esto se vuelve especialmente cierto en el caso de las tiendas, donde la gestión eficaz del inventario puede hacer la diferencia entre el éxito y el fracaso. Teniendo esto en mente, nuestro proyecto a continuación, se centra en el desarrollo de un programa de gestión de inventario (principalmente para tiendas mayoristas) elaborado con el lenguaje de programación C++.

El programa, denominado “Gestión de inventario”, es un sistema robusto y flexible diseñado para ayudar a las tiendas a manejar de una forma más eficaz y rápida su inventario o stock, además, siendo más específicos, “Gestión de inventario” permite a sus usuarios manejar el stock, realizar ventas y compras registradas, y generar informes detallados.

Además, C++, al ser ampliamente utilizado dentro de la industria de la programación y posee una gran comunidad de desarrolladores, el programa tendrá su propio mantenimiento cada cierto tiempo al igual que soluciones de errores espontáneos o nuevas añadiduras.

Resumiendo, este proyecto representa una oportunidad para todo tipo de tiendas que tengan un manejo arcaico de su inventario, y que por lo mismo, pierdan tiempo, dinero e interés, solucionando de esta manera un error sumamente significativo y muchas veces dejando desapercibido, y más si se trata de una tienda mayorista o distribuidora.

Definición y alcance del caso de estudio.

El caso de estudio se centra en el desarrollo de un programa de gestión de inventario para cualquier tipo de tiendas, denominado como “Gestión de inventario”, utilizando el lenguaje de programación C++. Este programa tiene como objetivo proporcionar una solución eficiente, precisa y rápida para el seguimiento y el manejo de inventario dentro de un entorno de tienda.

Alcance del Caso de Estudio.

El alcance de este caso de estudio abarca varios aspectos clave:

Desarrollo de Software: El proyecto implica un diseño, desarrollo e implementación de un programa de gestión de inventario utilizando C++. Incluyendo una implementación de funciones para el seguimiento y la gestión del inventario, y la integración de sistemas de base de datos para el almacenamiento de los mismos.

Gestión de Inventario: El programa permitirá al usuario realizar una variedad de tareas relacionadas con la gestión de inventario, incluyendo la gestión de stock, la realización y registro de compras y ventas, y la generación de informes detallados.

Aplicación Práctica: Tiene aplicaciones prácticas significativas. Las tiendas, de cualquier tipo (principalmente viéndose beneficiadas todas aquellas mayoristas o distribuidoras), podrán utilizar el programa para mejorar su eficiencia operativa y logística dentro del manejo de su inventario.

Actividades de la práctica de familiarización

Actividad 1: Descripción del **problema**/necesidad o caso de estudio.

En el mundo comercial actual, la gestión eficiente de un inventario es un componente crucial para el éxito de cualquier tienda o negocio más amplio. Sin embargo, muchas de estas todavía dependen de sistemas arcaicos y toscos, que, a menudo, son ineficientes, propensos a errores y/o requieren una gran cantidad de tiempo y esfuerzo manual para mantenerse en pie. Además, estos sistemas a menudo no se consideran una prioridad, lo que puede llevar a problemas de gestión de inventario más graves que pueden afectar negativamente a la rentabilidad y eficiencia de la tienda.

El problema se agrava aún más por el hecho de que muchos de estos sistemas antiguos no están diseñados para adaptarse a las cambiantes necesidades y demandas de las tiendas modernas. No pueden manejar grandes volúmenes de datos de productos, carecen de funcionalidades esenciales como la generación de informes, y no están integrados con otros sistemas comerciales esenciales. Como resultado, las tiendas frecuentemente se encuentran luchando con niveles de stock incorrectos, pérdida de ventas debido a la falta de productos, y una gran cantidad de tiempo y recursos desperdiciados en la gestión manual del mismo.

En respuesta a este problema, nuestro proyecto se centra en el desarrollo de un sistema de gestión de inventario más eficaz y mejor elaborado en C++, llamado “Gestión de inventario”. Este sistema está diseñado para abordar las deficiencias de los antiguos sistemas y proporcionar una solución moderna, eficiente y sencilla.

“Gestión de inventario” utiliza el lenguaje de código C++ y la flexibilidad que el mismo ofrece, para proporcionar una solución de inventario que es rápida, precisa y capaz de manejar datos correctamente. Con características como el manejo de stock, la realización de ventas y compras, o la generación de historiales, “Gestión de inventario” permite a las tiendas gestionar su inventario de manera más eficiente y efectiva.

Además, está diseñado para ser fácil de usar, lo que permite a las tiendas centrarse en lo que mejor saben hacer: vender productos y satisfacer a los clientes.

Esperamos de este proyecto abordar un problema real y significativo en el mundo comercial y proporciona una solución práctica y efectiva, mejorando la eficiencia y precisión de los inventarios en las tiendas.

Lista de Requerimientos funcionales.

Ingresar el nombre del producto.

Ingresar la cantidad del producto anteriormente añadido.

Seleccionar el producto a vender o solicitar ver su cantidad en stock.

Seleccionar la cantidad del producto seleccionado a vender.

Retirar producto totalmente del stock.

Realizar la compra con la cantidad correcta dicha por el comprador.

Verificar la cantidad restante de los productos anteriormente vendidos.

Revisar el historial anteriormente guardado.

Formulario de Google forms. (Estilo Rombo: Cerradas, Abiertas, Cerradas)

The image shows a Google Form titled "Gestion de Inventario" (Inventory Management). The form is set to "Recopilación de datos" (Data collection). At the top, it shows the email "crodrigueza@uamv.edu.ni" with a "Cambiar cuenta" (Change account) link and a "Se guardó el borrador" (Draft saved) notification. A red asterisk indicates that questions marked with an asterisk are mandatory. The form contains several sections: 1. "Correo electrónico *" (Email *): A checkbox is checked, indicating that "crodrigueza@uamv.edu.ni" will be included in the response. 2. "Nombres *" (Names *): A text field containing "Cristopher Amaru". 3. "Apellidos" (Surnames): A text field containing "Rodriguez Arauz". 4. A concluding instruction: "Se le solicita al usuario meterse en el papel de gerente de una tienda, ahora teniendo esto en cuenta, por favor responda a las siguientes preguntas dentro del mismo rol." (The user is requested to put themselves in the role of a store manager, now taking this into account, please answer the following questions within the same role.)

¿Qué tipos de productos o artículos ofrece su tienda? *

- ☒ Productos relacionados con el mundo del deporte
- ☐ Artículos relacionados con el mundo del maquillaje
- ☐ Artículos relacionados con el mundo de la moda
- ☐ Productos relacionados con el mundo de la limpieza dentro del hogar
- ☐ Otros: _____

¿Cuál crees que es el mayor desafío que has enfrentado/se puede enfrentar al gestionar el inventario de una tienda? *

- ☐ Lidar con el exceso o falta de stock de la mayoría de productos
- ☒ Lidar con la falta repentina de demanda de artículos anteriormente en auge
- ☐ Otros: _____

¿Cuál crees que sea el mejor sistema de gestión de inventario de una tienda? *

- ☐ Sistema de inventario manual
- ☒ Sistema de inventario digital pagado
- ☐ Sistema de inventario digital gratuito o pirata (con limitaciones o riesgos de virus)
- ☐ Otros: _____

¿Cómo manejarías el exceso de inventario o la falta del mismo dentro de tu tienda? *

Intentando revenderlos a precios menores y al por mayor para sacar el mayor provecho posible.

¿Cómo manejarías la pérdida de productos y dinero por confusiones y/o errores de inventario? *

Corrigiendo el problema desde la raíz y cambiándolo por un sistema que si sea total y completamente funcional.

¿Cómo intentarías recompensar la nula ganancia o pérdida de ingresos de la tienda? (por malas estrategias de tí y tu equipo) *

Intentaría recompensarlo teniendo mas en cuenta las opiniones de personas ajenas a mi equipo laboral para tener nuevas ideas frescas

Teniendo en cuenta que su tienda ha estado perdiendo dinero por el uso de un sistema de inventario digital antiguo y poco eficaz, cual cree que sea la mejor opción a tomar. *

- ☐ Migrar de sistema a uno manual con mas libertad pero mas riesgo de perdida de productos
- ☒ Migrar a un sistema digital actual mucho mejor y eficaz aunque tenga que capacitar a los trabajadores
- ☐ Mantener el sistema actual y acostumbrarse a sus errores para asi evitarlos
- ☐ Otros: _____

Si se le ofreciera un sistema digital, gratuito y eficaz, pero sabiendo que proviene de un grupo de personas con nula reputación dentro de internet. ¿Lo usaria dentro de su tienda?

- ☒ Si
- ☐ No
- ☐ Otros: _____

Borrar la selección

Enviar

Página 1 de 1

Borrar formulario

Nunca envíes contraseñas a través de Formularios de Google.

El formulario se creó en Universidad Americana. [Denunciar abuso](#)

Google Formularios

Diseño de entrada de datos.

Inicio:

¿Qué producto desea añadir?

¿Cuál es la cantidad del producto que desea añadir?

Actualización:

¿Qué producto va a añadir? (En caso de un nuevo producto)

Seleccione el producto deseado a modificar su cantidad

¿Cuál es la cantidad de productos que desea sumar al stock?

¿Cuál es la cantidad que desea retirar del stock? (En caso de pérdida de productos por cualquier motivo externo a una venta)

¿Qué producto desea retirar del stock? (En caso de que un artículo fuera descontinuado de la tienda)

Venta:

¿Qué producto desea vender?

¿Cuál es la cantidad que desea retirar?

Salida:

Recibo e Informe de la venta y el historial.

Actividad 2: **Análisis** del problema

Para gestionar eficazmente un inventario, se requieren ciertas entradas, una salida deseada específica, un método para producir la salida deseada y algunos requisitos adicionales y restricciones. Las entradas necesarias incluyen el código del producto, el nombre del producto, la cantidad, el precio, las opciones para agregar, borrar o buscar un producto, y la inversión en cada producto. La salida deseada es la capacidad de crear productos ingresando su nombre, precio y cantidad, y asegurando que estos datos se puedan guardar de manera persistente. Para producir esta salida deseada, el sistema debe emplear métodos que aseguren la persistencia de los datos, guardando toda la información ingresada y las operaciones realizadas en el disco duro de la computadora, garantizando que los datos permanezcan disponibles y no se eliminen al cerrar el programa. Además, el sistema debe cumplir con ciertos requisitos adicionales, como la capacidad de generar un informe detallado de todo lo disponible en el inventario, listando cada producto con su código, nombre, cantidad, precio y la inversión total. Este informe es esencial para revisiones y la planificación estratégica del inventario. El sistema también debe manejar grandes volúmenes de datos sin comprometer el rendimiento y ser fácil de usar, con una interfaz que permita a los usuarios realizar operaciones de manera rápida y eficiente. Además, es necesario asegurar la integridad de los datos mediante copias de seguridad regulares y la implementación de medidas de seguridad para proteger la información sensible del inventario contra accesos no autorizados.

R#1: Por cada sub-problema identificado, un análisis detallado utilizando la **matriz para análisis de problemas** proporcionada en clases.

Tipo de entrada	identificador	tipo de dato	valor inicial	dominio/rango	restriccion 1	restriccion 2
Conocido	balon de futbol	Entero	450 C\$	reales	≥ 0	
Conocido	uniforme de futbol	Entero	500 C\$	reales	≥ 0	
Conocido	balon de basketball	Entero	450 C\$	reales	≥ 0	
Conocido	uniforme de basketball	Entero	900 C\$	reales	≥ 0	
Conocido	pelota de beisbol	Entero	200 C\$	reales	≥ 0	
Conocido	Ingreso de producto	Entero		real	No se aceptan numeros	

descripcion	formulas
Programa(función)donde se llamara a las otras funciones Poder ingresar un producto al inventario	Ingreso de producto=añadir producto seleccionado + precio

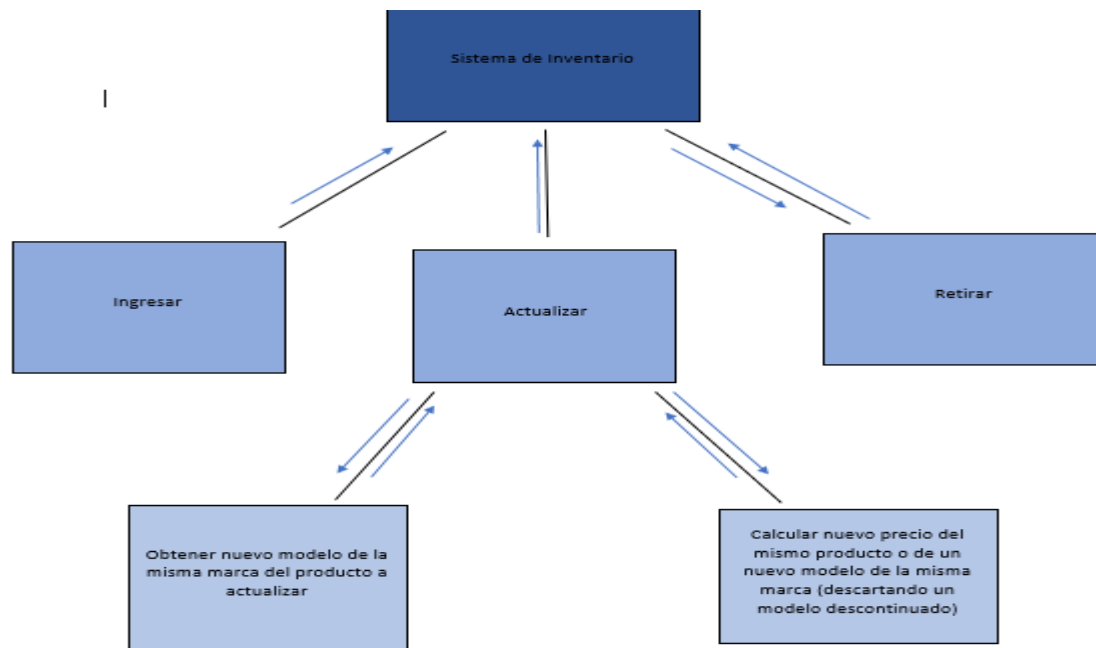
datos	resultados/formatos
Ingreso	Ingreso de un nuevo producto en el inventario (+1)

descripcion	formulas
Programa(función)donde se llamara a las otras funciones Actualizar cualquier productor dentro del inventario	actualizar producto= seleccionar un producto + precio/modelo a cambiar

datos	resultados/formatos
Actualización	Actualización de un producto dentro del inventario

descripcion	formulas
Programa(función)donde se llamara a las otras funciones Retiro de un objeto del inventario	retiro de objeto= seleccionar un producto + eliminar
datos	resultados/formatos
Retiro	Retiro de un producto del inventario
descripcion	formulas
Programa(función)donde se llamara a las otras funciones Ingresar nuevo modelo para reemplazar uno ya existente	reemplazar modelo= seleccionar objeto + añadir nuevo modelo + actualizar y reemplazar
datos	resultados/formatos
Nuevo modelo	Ingreso de nuevo modelo reemplazando el antiguo
descripcion	formulas
Programa(función)donde se llamara a las otras funciones Calcular nuevo precio de un producto ya existente en el inventario	cio= Seleccionar producto + eliminar precio anterior + añadir nuevo precio ya calculado ante
datos	resultados/formatos
Nuevo precio	Nuevo precio de un producto actualizado

R#2: El **diagrama de estructura** del programa.



Actividad 3: Diseño del algoritmo

1. Crear Producto:

Esta función permite añadir nuevos productos al inventario, asegurando que se registren adecuadamente todos los detalles necesarios para su identificación y gestión. Al crear un producto, se debe proporcionar un identificador único (como un código o número de referencia), el nombre del producto, la cantidad disponible, el precio por unidad y, en algunos casos, una descripción detallada del producto. Esta información es fundamental para el control y seguimiento del inventario, facilitando la identificación rápida y precisa de cada artículo. La creación de productos es esencial para mantener el inventario actualizado con nuevos artículos que ingresan al sistema, asegurando que estén listos para su gestión y venta.

2. Actualizar Producto:

Esta función permite modificar los detalles de un producto existente en el inventario. Es crucial para reflejar cambios que ocurren con el tiempo, como ajustes en la cantidad disponible debido a ventas o recepciones de mercancía, cambios en el precio debido a fluctuaciones del mercado, o actualizaciones en la descripción del producto para incluir nuevas características o especificaciones. La capacidad de actualizar productos garantiza que la información del inventario sea precisa y esté al día, lo cual es vital para la toma de decisiones, planificación de inventarios y satisfacción del cliente. Esta función asegura que todos los cambios relevantes se registren correctamente, manteniendo la integridad y precisión del inventario.

3. Borrar Producto:

Esta función permite eliminar productos específicos del inventario, lo cual es útil en situaciones donde un producto ya no está disponible, ha sido discontinuado o debe ser removido por otras razones. Eliminar un producto implica asegurarse de que se borren todos los registros asociados sin afectar negativamente a otros datos del inventario. En algunos sistemas, se mantiene un

registro de los productos eliminados para auditorías o análisis futuros. La eliminación de productos ayuda a mantener un inventario limpio y libre de artículos obsoletos o irrelevantes, mejorando la eficiencia operativa y la precisión en la gestión del inventario.

Actividad 4:

4. **Buscar Producto:**

Esta función permite localizar y obtener información detallada sobre un producto específico en el inventario utilizando varios criterios de búsqueda, como el identificador único, nombre del producto o categorías. Al buscar un producto, se puede acceder a información esencial como la cantidad disponible, el precio, la descripción y cualquier otra característica relevante. Esta capacidad de búsqueda es fundamental para la gestión eficiente del inventario, especialmente en sistemas con un gran número de productos. Facilita la identificación rápida de artículos, apoyo en la toma de decisiones, y mejora el servicio al cliente al proporcionar información precisa y accesible sobre los productos.

5. **Mostrar Todos los Productos Disponibles:**

Esta función genera una lista completa de todos los productos actualmente disponibles en el inventario. La lista incluye información clave como el identificador único de cada producto, su nombre, la cantidad disponible y el precio. Mostrar todos los productos disponibles proporciona una visión general del inventario, lo cual es útil para revisiones periódicas, auditorías y planificación de inventarios. Esta función permite a los administradores del inventario evaluar el estado actual de los artículos, identificar tendencias en las existencias y tomar decisiones informadas sobre reabastecimiento, promoción y discontinuación de productos.

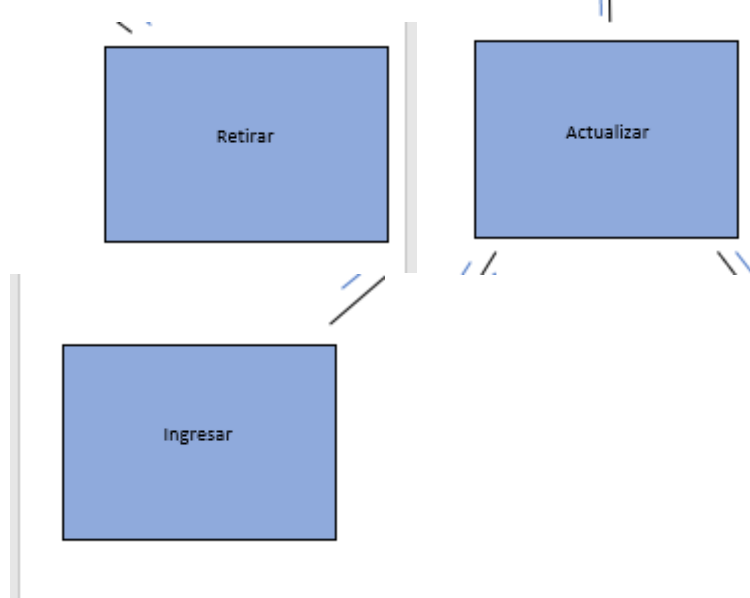
6. **Salir y Guardar un Resumen:**

Esta función se utiliza para cerrar el sistema de inventario de manera ordenada y guardar un resumen de todas las actividades realizadas durante la sesión. Al salir, el sistema registra todas las acciones como la creación, actualización, eliminación y búsqueda de productos, asegurando que todos los cambios se guarden correctamente. Este proceso de salida y guardado es vital para la integridad de los datos del inventario, proporcionando un registro detallado de todas las actividades para futuras referencias y auditorías. Generar informes resumen de las acciones realizadas facilita la evaluación del desempeño del inventario y ayuda en la planificación estratégica y la toma de decisiones informadas.

Estas funciones forman el núcleo de un sistema de gestión de inventarios eficiente, permitiendo un control riguroso y preciso de los productos, optimizando la administración de recursos y mejorando la toma de decisiones empresariales

Resultados de la actividad:

R#2: Un módulo por cada sub-programa diseñado



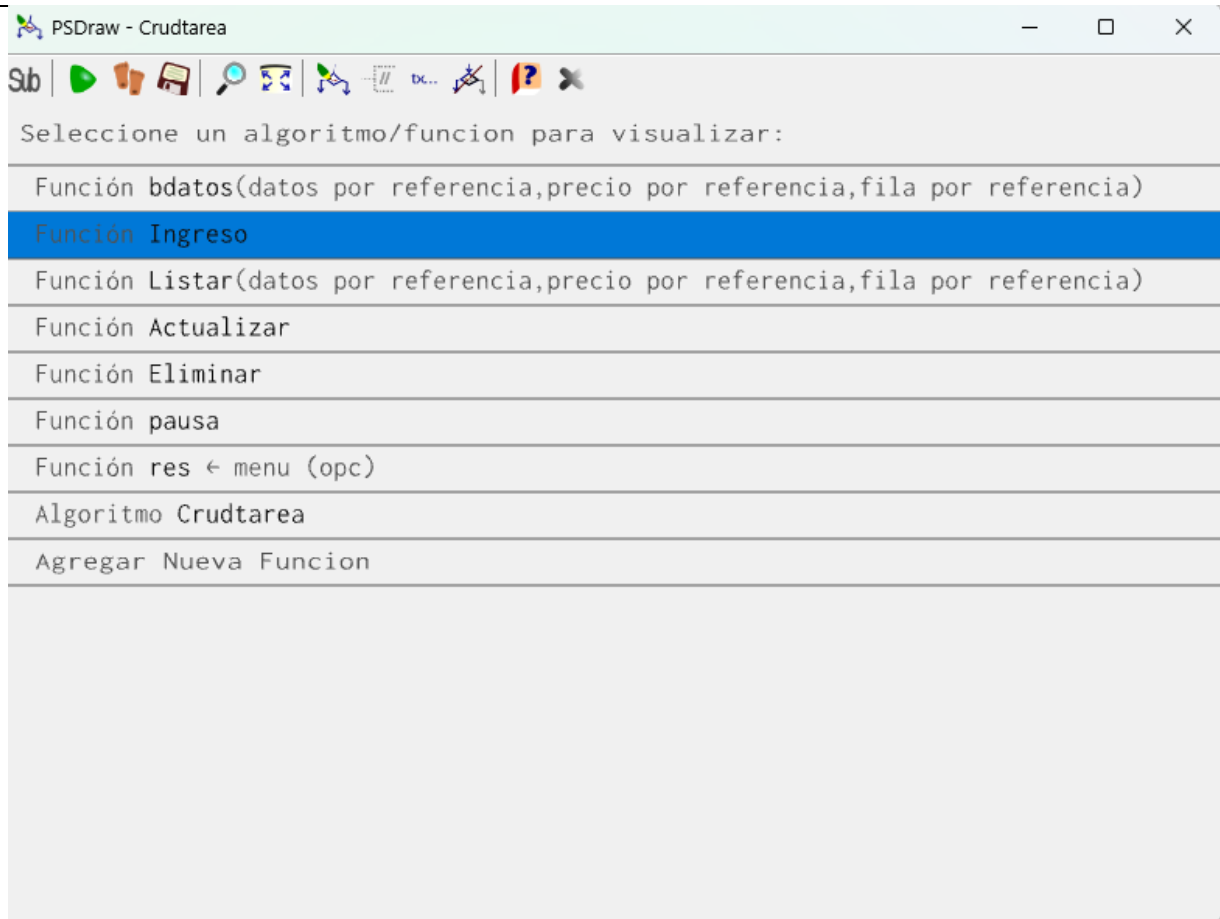
PseudoCodigo

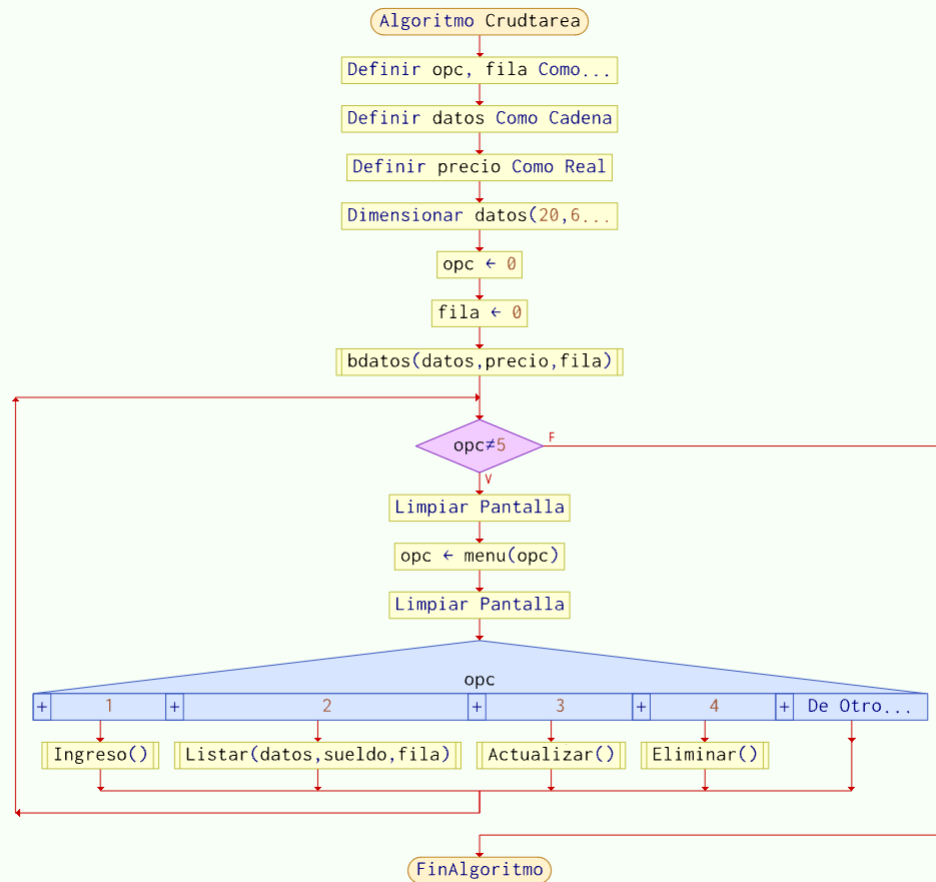
```
1  subproceso bdatos(datos por referencia, precio por referencia, fila por referencia)
2      datos[1,1] ← "0001";
3      datos[1,2] ← "Pelota de futbol";
4      datos[1,3] ← "C$350";
5
6      datos[2,1] ← "0002";
7      datos[2,2] ← "Guantes de portero";
8      datos[2,3] ← "C$450";
9
10     datos[3,1] ← "0003";
11     datos[3,2] ← "Pelota de Basketball";
12     datos[3,3] ← "C$500";
13
14     fila ← 4;
15 FinSubProceso
16
17
18 subproceso Ingreso
19     Escribir "Registrar";
20
21     pausa()
22 Finsubproceso
23
24 subproceso Listar(datos por referencia, precio por referencia, fila por referencia)
25     Definir i, j Como Entero
26     Escribir "Codigo Producto      Precio      ";
27     Escribir "-----";
28     Para i ← 1 Hasta fila-1 Hacer
29         Para j ← 1 Hasta 4 Hacer
30             Escribir Sin Saltar datos[i,j], " ";
31         FinPara
32         Escribir precio[i];
33     FinPara
34
35
36     pausa()
37 Finsubproceso
38
39 Función Actualizar
40 Escribir "Actualizar";
41
42     pausa()
```

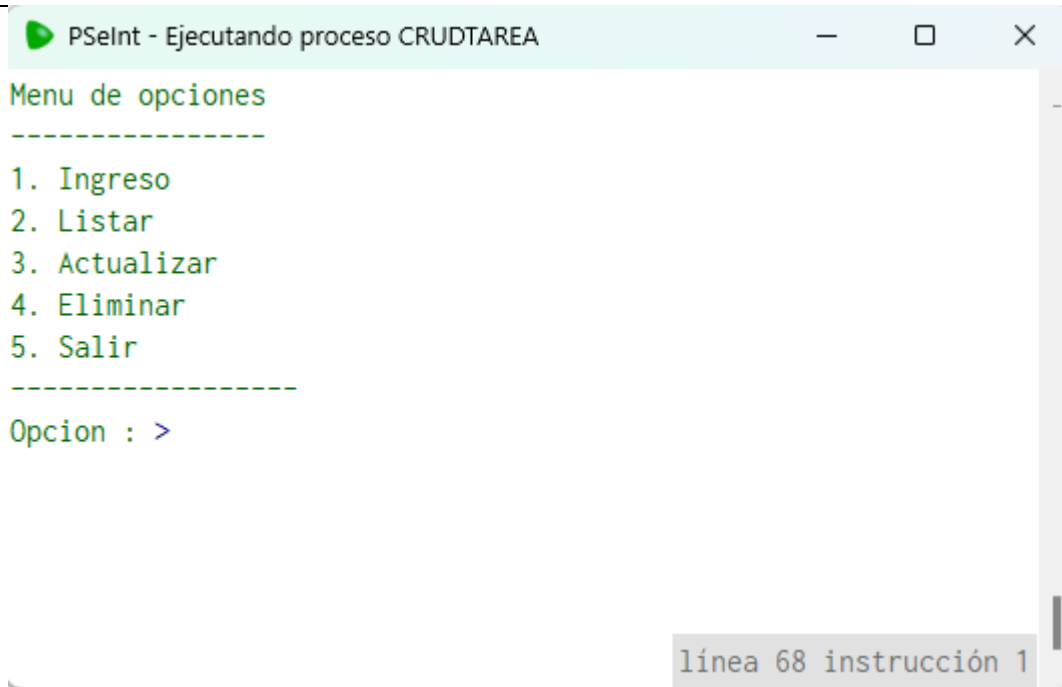
```
42     pausa()
43 FinFunción
44
45 Función Eliminar
46     Escribir "Eliminar";
47
48     pausa()
49 FinFunción
50
51 Función pausa
52     Escribir ""
53     Escribir "Pulsar una tecla para continuar"
54     Esperar Tecla
55 FinFunción
56
57 Función res <- menu (opc)
58     Definir res Como Entero
59     Escribir "Menu de opciones"
60     Escribir "-----"
61     Escribir "1. Ingreso"
62     Escribir "2. Listar"
63     Escribir "3. Actualizar"
64     Escribir "4. Eliminar"
65     Escribir "5. Salir"
66     Escribir "-----"
67     Escribir "Opcion : " Sin Saltar
68     Leer res
69 FinFunción
70
71 Algoritmo Crudtarea
72     Definir opc, fila Como Entero
73     Definir datos Como Cadena
74     Definir precio Como Real
75     Dimensionar datos[20,6], sueldo[20]
76
77     opc ← 0;
78     fila ← 0;
79
80     bdatos(datos, precio, fila);
81     Mientras opc <> 5 Hacer
82         Limpiar Pantalla;
83         opc ← menu(opc);
```



```
77     opc ← 0;
78     fila ← 0;
79
80     bdatos(datos, precio, fila);
81     Mientras opc <> 5 Hacer
82         Limpiar Pantalla;
83         opc ← menu(opc);
84         Limpiar Pantalla;
85         Según opc Hacer
86             1:Ingreso();
87             2:Listar(datos, sueldo, fila);
88             3:Actualizar();
89             4:Eliminar();
90         FinSegún
91     FinMientras
92 FinAlgoritmo
```

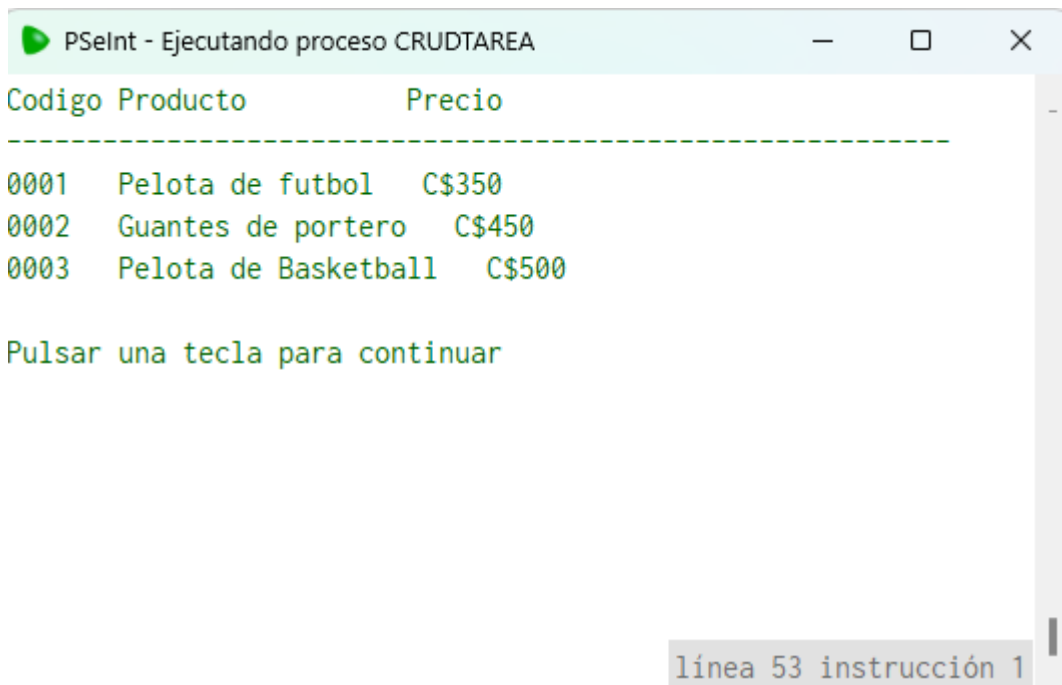






```
PSeInt - Ejecutando proceso CRUDTAREA
Menu de opciones
-----
1. Ingreso
2. Listar
3. Actualizar
4. Eliminar
5. Salir
-----
Opcion : >
```

línea 68 instrucción 1

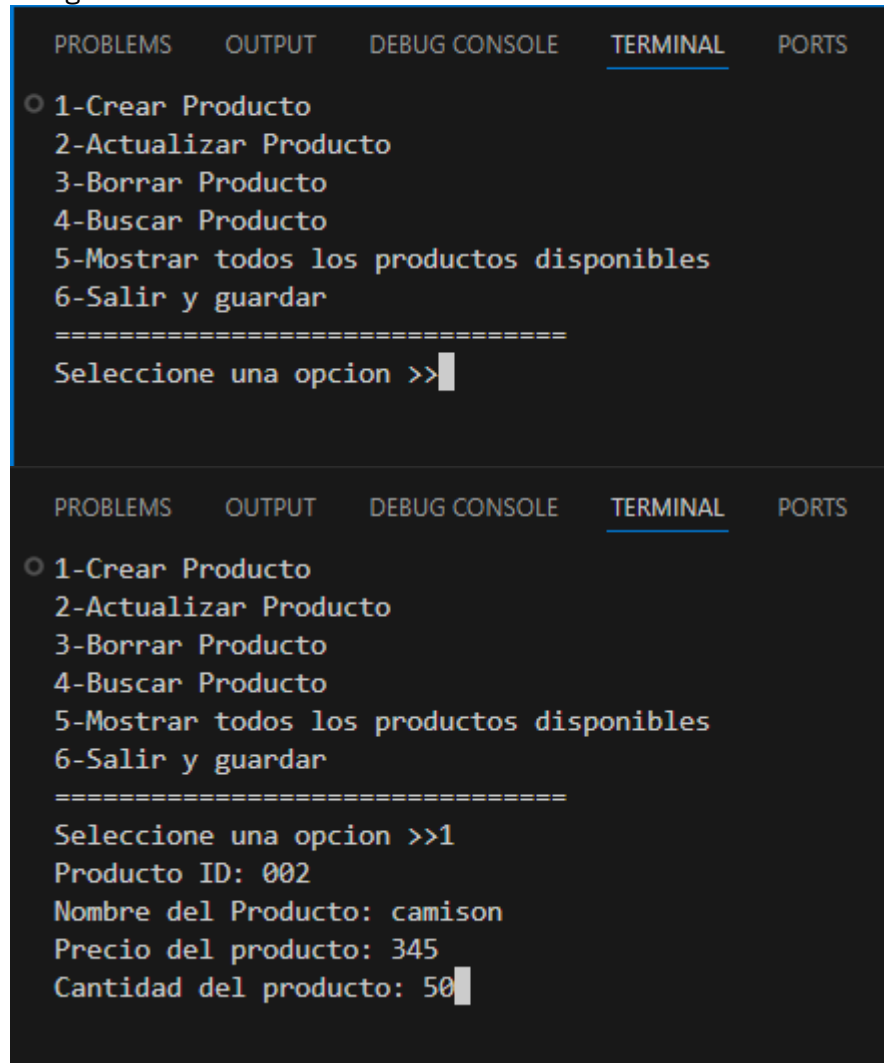


```
PSeInt - Ejecutando proceso CRUDTAREA
Codigo Producto      Precio
-----
0001  Pelota de futbol  C$350
0002  Guantes de portero C$450
0003  Pelota de Basketball C$500

Pulsar una tecla para continuar
```

línea 53 instrucción 1

Codigo resultados



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
○ 1-Crear Producto
  2-Actualizar Producto
  3-Borrar Producto
  4-Buscar Producto
  5-Mostrar todos los productos disponibles
  6-Salir y guardar
=====
Seleccione una opcion >>

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
○ 1-Crear Producto
  2-Actualizar Producto
  3-Borrar Producto
  4-Buscar Producto
  5-Mostrar todos los productos disponibles
  6-Salir y guardar
=====
Seleccione una opcion >>1
Producto ID: 002
Nombre del Producto: camison
Precio del producto: 345
Cantidad del producto: 50
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
○ Los Productos son
=====
No. | ID | Nombre | Precio | Cantidad | Monto
-----
1 002 camison 345 50 17250
=====
1-Crear Producto
2-Actualizar Producto
3-Borrar Producto
4-Buscar Producto
5-Mostrar todos los productos disponibles
6-Salir y guardar
=====
Seleccione una opcion >>

1-Crear Producto
2-Actualizar Producto
3-Borrar Producto
4-Buscar Producto
5-Mostrar todos los productos disponibles
6-Salir y guardar
=====
Seleccione una opcion >>3
Eliminar por ID >>002
Eliminacion completada!
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Los Productos son
=====
No. | ID | Nombre | Precio | Cantidad | Monto
-----
Producto no encontrado!
=====

1-Crear Producto
2-Actualizar Producto
3-Borrar Producto
4-Buscar Producto
5-Mostrar todos los productos disponibles
6-Salir y guardar
=====
Seleccione una opcion >>

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Los Productos son
=====
No. | ID | Nombre | Precio | Cantidad | Monto
-----
Producto no encontrado!
=====

1-Crear Producto
2-Actualizar Producto
3-Borrar Producto
4-Buscar Producto
5-Mostrar todos los productos disponibles
6-Salir y guardar
=====
Seleccione una opcion >>6
Saliendo... Archivo guardado!
* Terminal will be reused by tasks, press any key to close it.
```

Contacto.

Nombre completo: Harald Antonio Corea Salas

Teléfono: 5780 3402

Dirección: del kilómetro 7 y medio carretera sur frente a farma de descuento

Email: coreaharald17@gmail.com

Gerente del módulo de servicio al cliente del programa “Gestión de inventario”

Referencias Bibliográficas

Joyanes, L. (2008). Fundamentos de Programación. Algoritmos, estructura de datos y objetos. (4ta Ed.). Madrid: McGraw-Hill/Interamericana

Joyanes, L., Zahonero I. (2005) PROGRAMACIÓN EN C. Metodología, algoritmos y estructura de datos (2da Ed.) Madrid: McGraw-Hill/Interamericana
Maner, W. (27 de octubre de 2020). *PROTOTIPADO*. Obtenido de <http://www.sidar.org/recur/desdi/traduc/es/visitable/maner/Prototipado.htm>