



Trabajo de Final para el curso de Fundamentos de Programación

BOCINOVICH RUMAY AIRTON LEONARDO

Ingeniería de Sistemas

CAMA COSTILLO JOHNNY SEBASTIAN

Ingeniería de Software

POMA MEDINA DIEGO SEBASTIAN

Ingeniería de Sistemas

Profesor:

DIAZ ARENAS DANIE JESUS

LIMA – PERÚ

2023

Tabla de contenidos

Introducción	3
1 Descripción del problema computacional	3
2 Objetivo General	5
3 Objetivos específicos	5
4 Desarrollo del Aplicativo	6
4.1 Diseño de la interfaz o el uso de la consola	6
4.1.1 Menú Principal	6
4.1.2 Sistema de Cliente	7
4.1.3 Sistema de Vendedor	7
4.1.4 Interfaz del Menú de Productos	8
4.2 Desarrollo del producto	10
5 Conclusiones	10
6 Glosario de términos	12

Introducción

Este código implementa un sistema de ventas para una tienda de juguetes llamada VICTOYS. Está escrito en C++ y utiliza estructuras para representar clientes, vendedores y productos. El programa ofrece funcionalidades como registro y validación de clientes, registro de productos, operaciones de ventas, modificación y eliminación de productos, y generación de reportes de ventas.

La aplicación inicia con un menú principal que permite acceder a dos sistemas: el sistema de clientes y el sistema de vendedores (cajeros). Los clientes pueden registrarse, iniciar sesión y realizar operaciones relacionadas con la compra de productos. Por otro lado, los vendedores pueden registrar información sobre productos, gestionar ventas, modificar y eliminar productos, entre otras funciones.

En el sistema de productos, se proporciona un menú para seleccionar productos disponibles, agregar productos al carrito, buscar productos, listar productos en el carrito y salir. Cada producto tiene un código, nombre, precio y cantidad en stock. Además, el sistema de ventas permite realizar operaciones como la generación de una venta, validar la venta y obtener un reporte del estado de la venta.

El código se organiza en funciones y utiliza estructuras de datos como vectores para almacenar información sobre clientes, vendedores y productos. Además, se implementan bucles y condicionales para gestionar las distintas opciones y operaciones del sistema.

1 Descripción del problema computacional.

El código proporcionado implementa un sistema de ventas para una tienda de juguetes llamada VICTOYS. El sistema está dividido en dos módulos principales: el sistema de clientes y el sistema de cajeros. Aquí hay una descripción concisa de cada parte:

❖ Sistema de Clientes:

➤ **Registro de Clientes [regCliente]:**

- Permite a los clientes registrarse en el sistema proporcionando un nombre de usuario y una contraseña.
- Asigna automáticamente un código de cliente único.
- Almacena la información del cliente en un vector.

➤ **Validación de Clientes [valCliente]:**

- Permite a los clientes iniciar sesión mediante la validación del código, nombre de usuario y contraseña.
- Después de iniciar sesión, permite a los clientes realizar operaciones en el sistema de ventas.

➤ **Operaciones del Sistema de Ventas:**

- Permite a los clientes realizar acciones como registrar productos, listar productos y salir del sistema.

❖ **Sistema de Cajeros:**

➤ **Registro de Cajeros [regVendedor]:**

- Permite a los cajeros registrarse en el sistema proporcionando un número de caja y un nombre.

➤ **Menú del Cajero [menVendedor]:**

- Permite a los cajeros realizar diversas operaciones en el sistema de caja, como generar ventas, modificar productos y finalizar operaciones.

➤ **Operaciones del Sistema de Caja:**

- Permite a los cajeros realizar acciones como generar ventas, modificar productos, eliminar productos y finalizar operaciones de venta.

❖ **Sistema de Productos:**

➤ **Registro de Productos [regProducto]:**

- Permite a los usuarios seleccionar y agregar productos al carrito de compra.
- Cada producto tiene un código, nombre, precio y cantidad.

➤ **Funcionalidades Adicionales:**

- Buscar productos por código.
- Listar productos en el carrito.
- Modificar la cantidad de productos en el carrito.
- Eliminar productos del carrito.

❖ **Funcionalidades de Venta:**

➤ **Generación de Venta [generarVenta]:**

- Calcula el precio total de los productos en el carrito.

➤ **Realización de Venta [venta]:**

- Muestra los detalles de los productos en el carrito, incluyendo el precio total.

El sistema utiliza estructuras de datos como vectores para almacenar la información de clientes, cajeros y productos. Además, proporciona una interfaz de consola para la interacción del usuario.

2 Objetivo General

Desarrollar un sistema de gestión de ventas para una tienda de juguetes llamada VICTOYS, que permita a los clientes registrarse, acceder al sistema, agregar productos al carrito, realizar ventas, y a los vendedores gestionar su registro, realizar operaciones de venta, modificar y eliminar productos, generando un entorno interactivo y eficiente para el proceso de compra y venta de productos

3 Objetivos específicos

El código implementa un sistema de ventas para una tienda de juguetes llamada VICTOYS. Se manejan distintos sistemas, como el de clientes y el de vendedores, permitiendo el registro, login, y operación de compra y venta de productos.

- ❖ Registro y Validación de Clientes:
 - Permitir a los clientes registrarse en el sistema.
 - Validar el acceso de los clientes al sistema mediante un proceso de login.
- ❖ Registro y Operación de Cajeros:
 - Registrar cajeros en el sistema.
 - Permitir a los cajeros realizar diversas operaciones de venta y gestión de productos.
- ❖ Gestión de Productos:
 - Ofrecer un menú para registrar nuevos productos o seleccionar existentes.
 - Permitir la búsqueda, modificación y eliminación de productos.
- ❖ Carrito de Compras:
 - Mantener un carrito de compras para cada cliente, almacenando productos seleccionados.
 - Mostrar la lista de productos en el carrito y permitir la modificación y eliminación de productos.
- ❖ Generación de Ventas:
 - Calcular el precio total de la compra sumando los productos en el carrito.
 - Validar y concluir la venta, permitiendo la confirmación o cancelación por parte del cliente.
- ❖ Menú Principal:
 - Crear un menú principal que permita navegar entre los diferentes sistemas y opciones de la aplicación.
 - Proporcionar una opción para finalizar la aplicación.

4 Desarrollo del Aplicativo

4.1 Diseño de la interfaz o el uso de la consola

4.1.1 Menú Principal

```

-----
SISTEMA DE VENTAS DE VICTOYS
-----
1. Sistema de Clientes
2. Sistema de Vendedores
3. Salir
Ingrese una opcion[1-3]:

```

4.1.2 Sistema de Cliente

```
-----  
SISTEMA DE CLIENTE  
-----  
1. Registro  
2. Login  
3. Finalizar Registro De Clientes  
Ingrese una opcion[1-3]:
```

```
-----  
REGISTRO DE CLIENTES  
-----  
Codigo: [Autoincremental]  
Ingrese nombre de usuario: [Input]  
Ingrese contraseña: [Input]  
Cliente registrado exitosamente.  
-----
```

```
-----  
ACCESO AL SISTEMA  
-----  
Codigo: [Input]  
Usuario: [Input]  
Contraseña: [Input]  
Acceso al sistema correcto  
-----
```

4.1.3 Sistema de Vendedor

```
-----  
SISTEMA DEL CAJERO  
-----  
1. Registro De Personal  
2. Iniciar Operacion  
3. Finalizar Operacion  
Ingrese una opcion[1-3]:
```

```

-----
REGISTRO DE CAJA
-----

Número de Caja: [Ingrese número de caja]
Ingrese nombre del cajero: [Ingrese nombre del cajero]

```

```

-----
ACCESO AL SISTEMA
-----

Numero De Caja:
Usuario:
Acceso al sistema correcto
-----

SISTEMA DE CAJA
-----

Reporte De La Compra _____[1]
Modificacion De Producto_____ [2]
Eliminacion De Producto_____ [3]
Generar Venta_____ [4]
Validacion De La Venta_____ [5]
Salida De La Venta_____ [6]
Concluir Venta_____ [7]
Ingrese una opcion[1-7]:

```

4.1.4 Interfaz del Menú de Productos

```

-----
MENU DE REGISTRO DE PRODUCTOS DE VICTOYS
-----

1. Seleccion de Productos
2. Buscar Producto
3. Listar Productos
4. Salir
Ingrese una opcion[1-4]:

```



```

-----
PRODUCTOS DISPONIBLES DE VICTOYS
-----

```

1. MAX STEELL - \$25.30
2. OSO TED - \$12.50
3. BARBIE - \$28.70
4. HOT WHEELS - \$10.00
5. TRANSFORMER - \$16.80
6. CUBO MAGICO - \$5.00
7. GOGOS - \$3.40
8. LEGO - \$100.00
9. NERF - \$43.70
10. PLAY DOH - \$30.00

```

-----
Seleccione un producto[1-10]:

```

```

Ingrese el codigo a buscar:

```

```

-----
Cliente: [Nombre del Cliente] (Código: [Código del Cliente])
-----

```

```

-----
Codigo:[Código del Producto]

```

```

Nombre:[Nombre del Producto]

```

```

Precio:[Precio del Producto]

```

```

Cantidad:[Cantidad del Producto]

```

```

-----
[Información del siguiente producto]

```

```

...

```

4.2 Desarrollo del producto

```
#include<iostream>
#include<vector>
#include<string>
using namespace std;

// Estructuras de datos para Cliente, Vendedor y Producto
struct Cliente { /* ... */ };
struct Vendedor { /* ... */ };
struct Producto { /* ... */ };

// Vectores para almacenar clientes, vendedores y productos
vector<Cliente> vecCliente;
vector<Vendedor> vecVendedor;
vector<Producto> vecProducto;

// Variables para mantener el estado del cliente actual
string nombreClienteActual = "";
int codigoClienteActual = 0;

// Funciones prototipo
void sisCliente();
void regCliente();
int codCliente();
void valCliente();
void sisVendedor();
void regVendedor();
void menVendedor();
void regProducto();
void addProducto();
int getProducto();
void buscarProducto();
void listarProducto();
void modificarProducto();
int posicionProducto(int);
void eliminarProducto();
float generarVenta();
void venta();

// Función principal (main)
int main() {
    // ... (menú principal y bucle)
}

// Desarrollo de las funciones
// ...
```

5 Conclusiones

El código proporcionado es un programa en C++ que simula un sistema de ventas para una tienda de juguetes llamada VICTOYS. El sistema incluye la gestión de clientes, vendedores y

productos, permitiendo registrar clientes, realizar ventas, y gestionar el inventario de productos.

En resumen, el código realiza las siguientes funciones principales:

- Registro de Clientes: Permite registrar clientes con un código único, nombre de usuario y contraseña.
- Validación de Clientes: Permite a los clientes registrados iniciar sesión en el sistema.
- Registro de Vendedores: Permite registrar vendedores (cajeros) con un número único y nombre.
- Operaciones de Vendedor: Permite a los vendedores realizar operaciones como reportes de compra, modificación y eliminación de productos, generación de ventas, validación de ventas y salida de ventas.
- Registro de Productos: Permite agregar productos al inventario, cada uno con un código único, nombre, precio y cantidad en stock.
- Listado de Productos: Muestra la lista de productos en el inventario.
- Modificación y Eliminación de Productos: Permite a los vendedores modificar la cantidad de productos en el inventario o eliminar productos.
- Generación de Venta: Permite a los clientes agregar productos a su carrito de compras.
- Validación y Salida de Venta: Permite a los vendedores validar la venta y mostrar el estado de esta.

El código utiliza estructuras de datos como vectores para almacenar información de clientes, vendedores y productos. También incluye funciones para realizar diversas operaciones dentro del sistema. En general, el código proporciona una base funcional para un sistema de venta.

6 Glosario de términos

1. Estructuras:

- Cliente: Almacena información sobre un cliente, incluyendo su código, nombre de usuario y contraseña.
- Producto: Contiene detalles de un producto, como código, nombre, precio y cantidad en stock.
- Vendedor Guarda información de un vendedor, como número de caja y nombre.

2. Vectores:

- vecCliente: Vector que almacena objetos de tipo Cliente.
- vecVendedor: Vector que guarda objetos de tipo Vendedor.
- vecProducto Vector que contiene objetos de tipo Producto.

3. Funciones Principales:

- main(): Función principal que controla el flujo del programa y muestra el menú principal.
- sisCliente(): Gestiona el sistema relacionado con los clientes, con opciones como registro, login y finalización.
- regCliente(): Registra a un nuevo cliente solicitando nombre de usuario y contraseña.
- -valCliente(): Valida el acceso de un cliente mediante un código, usuario y contraseña.
- sisVendedor(): Maneja el sistema del cajero, con opciones para registro, inicio y finalización de operaciones.

- `regVendedor()`: Registra a un nuevo vendedor solicitando el número de caja y el nombre.
- `menVendedor()`: Gestiona las operaciones del sistema de caja, como registro de compra, modificación y eliminación de productos, generación de ventas y validación.

4. Funciones de Producto:

- `regProducto()`: Controla el menú de registro de productos con opciones para agregar, buscar, listar y salir.
- `addProducto()`: Permite agregar productos al carrito con opciones predefinidas.
- `getProducto()`: Genera el código único para un nuevo producto.
- `buscarProducto()`: Busca un producto por su código e imprime sus detalles.
- `listarProducto()`: Muestra la lista de productos en el carrito.
- `posicionProducto()`: Devuelve la posición de un producto en el carrito según su código.
- `modificarProducto()`: Permite modificar la cantidad de un producto en el carrito.
- `eliminarProducto()`: Elimina un producto del carrito.

5. Funciones de Venta:

- `generarVenta()`: Calcula el precio total de la venta sumando el precio de cada producto.
- `venta ()`: Muestra los detalles de la venta, incluyendo productos, cantidades y precio total.

6. Flujo de Control:

- `do-while`: Se utiliza para repetir el menú principal y otros menús secundarios hasta que se selecciona la opción de salida.

7. Entrada/Salida:

- `cout/cin`: Se utiliza para la entrada y salida de datos en la consola.
- `system("cls")`: Limpia la pantalla de la consola.
- `getline()`: Lee una línea completa, especialmente útil para la entrada de cadenas.

8. Variables Globales:

- `nombreClienteActual`: Almacena el nombre del cliente actual para su uso en diversas partes del programa.
- `codigoClienteActual`: Almacena el código del cliente actual para referencia en diferentes secciones del programa.

Este glosario proporciona una visión general de la estructura y funcionalidad del código, destacando términos clave y sus respectivas funciones.