

Situación 4: El Problema del Cajero Automático

¿Qué tipo de programa necesitas crear? Un sistema de seguridad para cajeros automáticos con autenticación de PIN, control de intentos y gestión de sesiones.

¿Qué validaciones son necesarias?

- Validación de PIN (4 dígitos numéricos)
- Verificación de intentos fallidos consecutivos
- Validación de saldo suficiente para transacciones
- Control de tiempo de sesión activa
- Verificación de estado de la cuenta (activa/bloqueada)

¿Qué casos especiales debes considerar?

- Bloqueo automático después de múltiples intentos fallidos
- Tiempo límite de inactividad
- Recuperación de sesión en caso de error del sistema
- Transacciones parcialmente completadas
- Ataques de fuerza bruta al PIN

¿Qué entrada necesita el programa del usuario?

- Número de tarjeta o cuenta
- PIN de 4 dígitos
- Tipo de transacción deseada
- Monto de la transacción (si aplica)

¿Qué debe mostrar como resultado?

- Estado de autenticación (exitosa/fallida)
- Saldo disponible
- Menú de opciones disponibles
- Confirmación de transacciones
- Mensajes de error específicos
- Tiempo restante de sesión

CÓDIGO

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

```
int main() {
```

```
const int Intentos_Maximos = 10;
const int Clave_Correcta = 1234;
int CLAVE, Intentos = 0, saldo = 1000;
bool acceso = false;
```

```
while (Intentos < Intentos_Maximos && !acceso) {
    cout << "Ingresa tu PIN (4 digitos): ";
    cin >> CLAVE;
    if (CLAVE == Clave_Correcta) {
        acceso = true;
    } else {
        Intentos++;
        cout << "PIN incorrecto. Intentos restantes: " << (Intentos_Maximos -
Intentos) << endl;
        if (Intentos == Intentos_Maximos) {
            cout << "Demasiados intentos. Cajero bloqueado." << endl;
            return 0;
        }
    }
}
```

```
int opcion;
while (acceso) {
    cout << "\nMenu:\n1. Retirar dinero\n2. Ver saldo disponible\n3. Salir\nElige
una opcion: ";
    cin >> opcion;
```

```
switch (opcion) {
    case 1: {
        int retiro;
        cout << "Saldo disponible: $" << saldo << endl;
        cout << "Cantidad a retirar: $";
        cin >> retiro;
        if (retiro <= saldo) {
            cout << "Estas seguro de retirar $" << retiro << "? (1 = Si, 0 = No): ";
            int confirm;
            cin >> confirm;
            if (confirm == 1) {
                saldo -= retiro;
                cout << "Retiro exitoso. Saldo restante: $" << saldo << endl;
            } else {
                cout << "Retiro cancelado." << endl;
            }
        }
    }
```

```
    }  
    } else {  
        cout << "Saldo insuficiente." << endl;  
    }  
    break;  
}  
case 2:  
    cout << "Saldo disponible: $" << saldo << endl;  
    break;  
case 3: // Salir  
    cout << "Gracias por usar el cajero. Adios." << endl;  
    acceso = false;  
    break;  
default:  
    cout << "Opcion invalida." << endl;  
}  
}  
  
return 0;  
}
```

DIAGRAMA DE FLUJO

