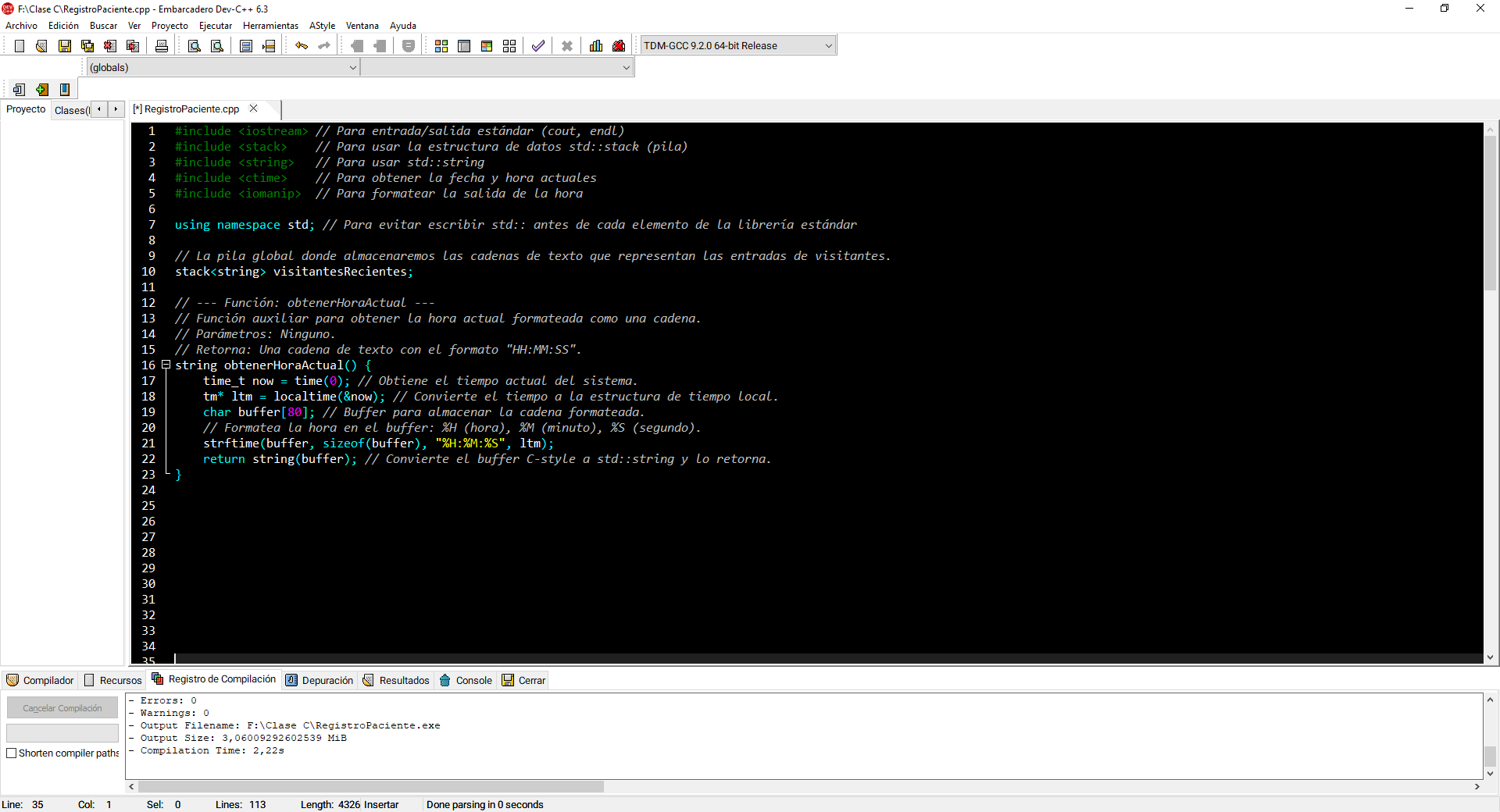
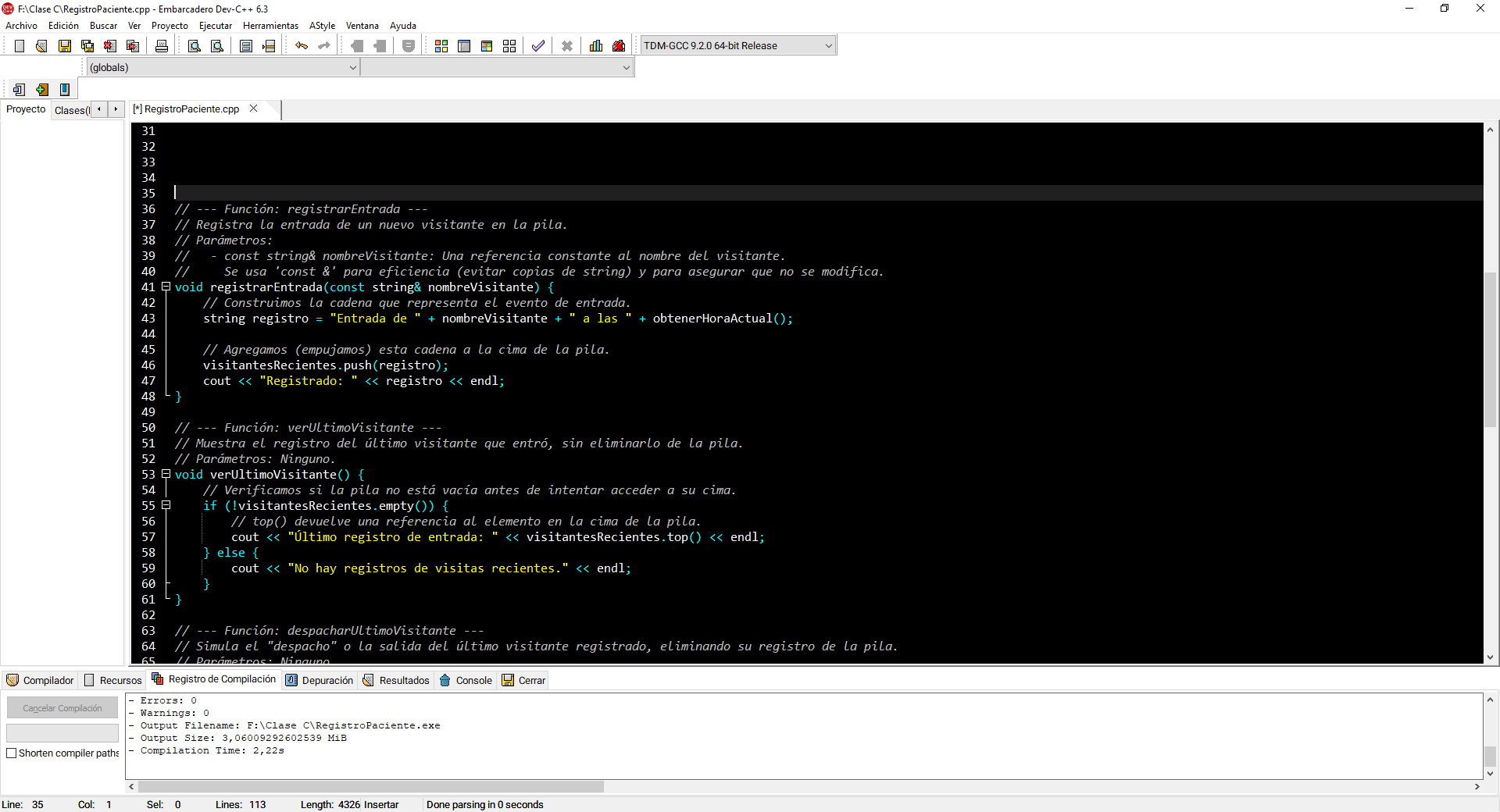
**Registro de Visitas Recientes**

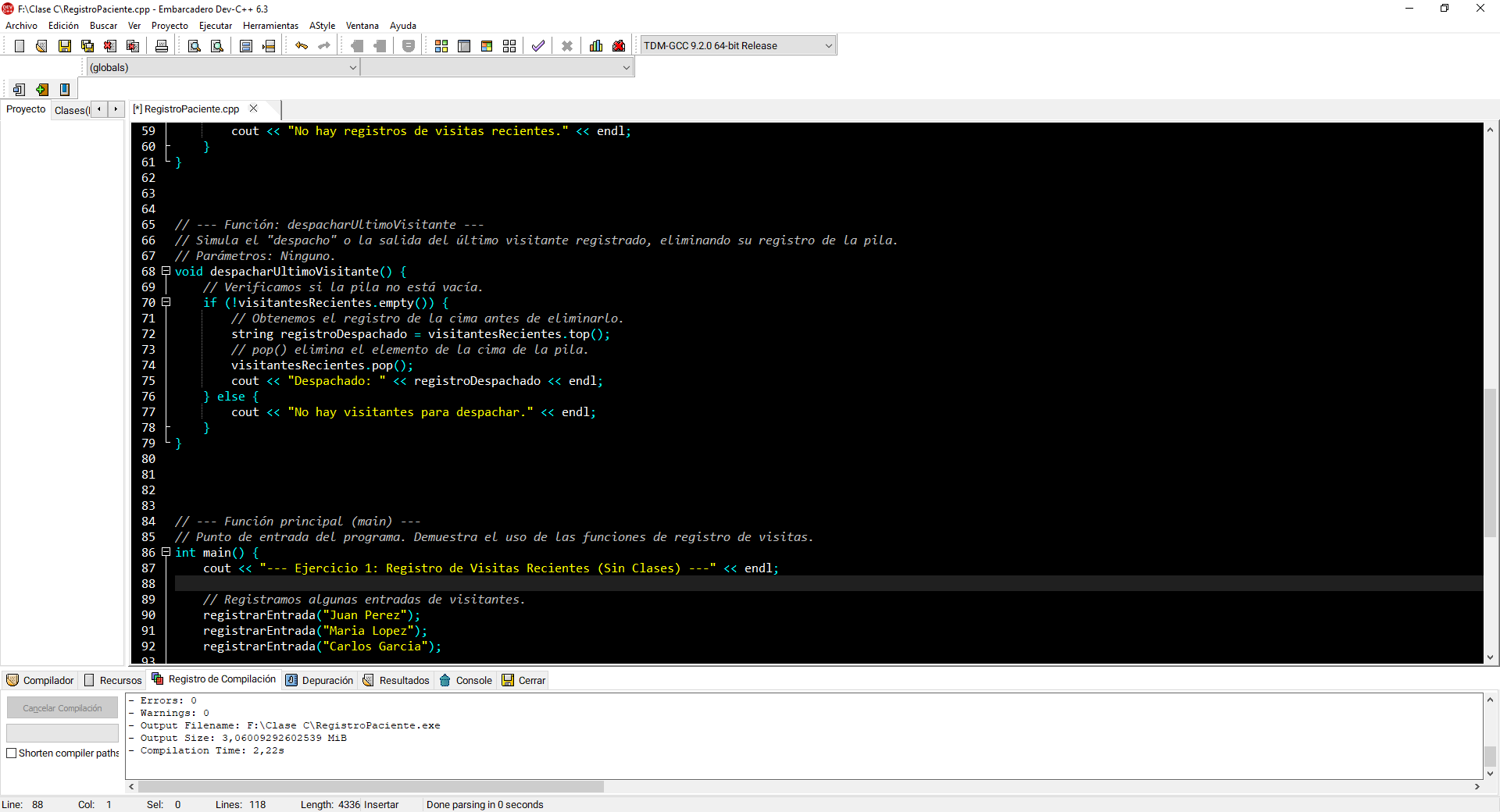
**Descripción del Problema:** Imagina un sistema de control de acceso a una oficina donde quieres saber quién fue la última persona en entrar. La pila es perfecta para esto, ya que siempre te dará el último registro.

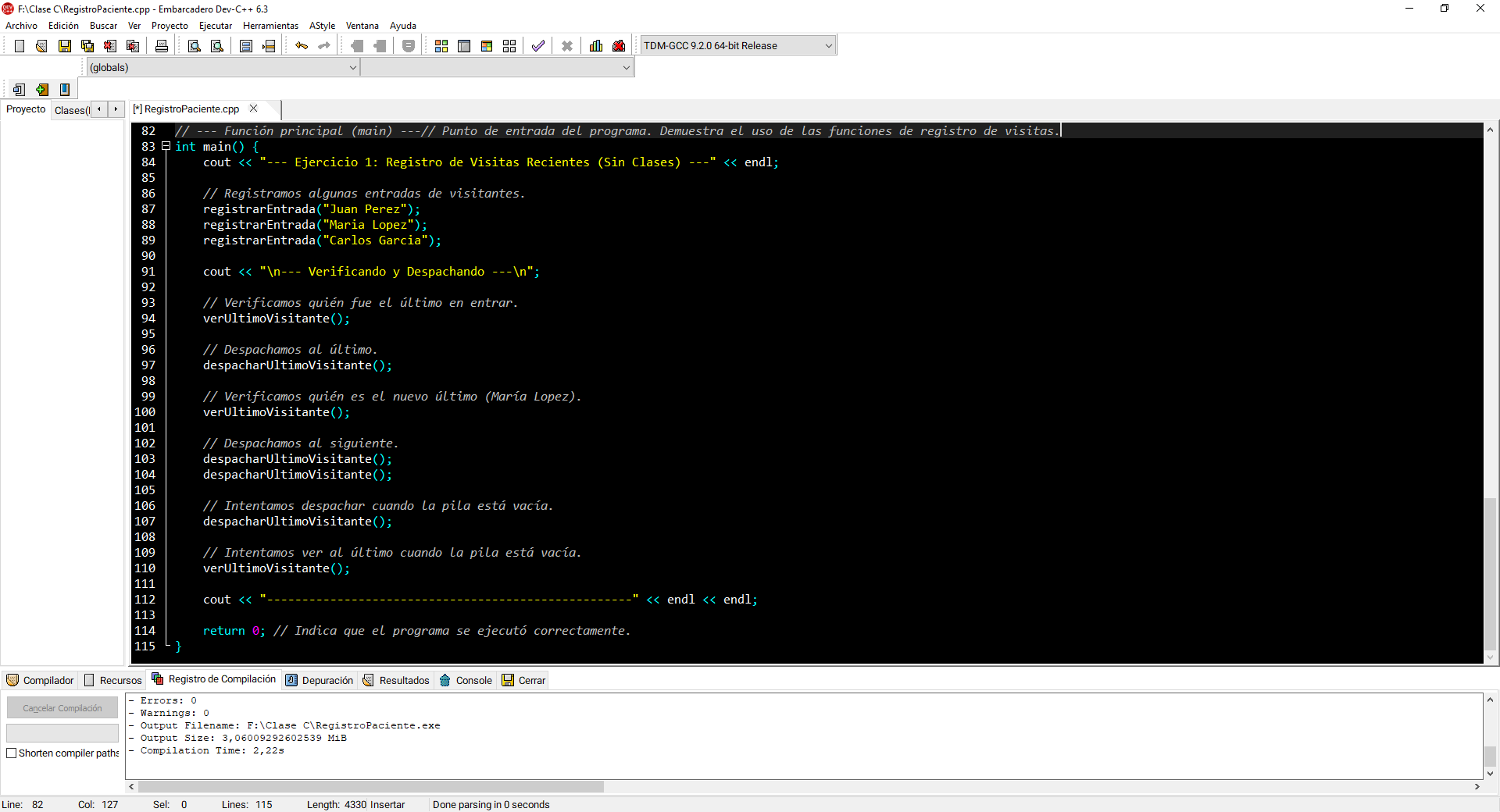
**Requisitos Técnicos:**

1. Definir una estructura Visitante con un nombre y una hora de entrada.
2. Usar una **pila** para almacenar los visitantesRecientes.
3. Implementar una función para registrarEntrada(Visitante) que agregue un visitante a la pila.
4. Implementar una función para verUltimoVisitante() que muestre quién fue el último en entrar sin quitarlo de la pila.
5. Implementar una función para despacharUltimoVisitante() que retire al último visitante y lo muestre.

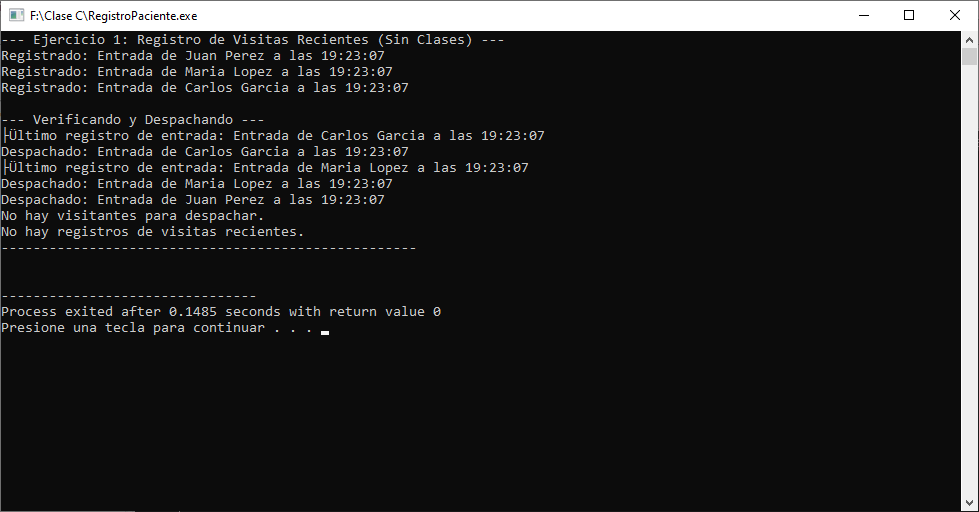








Salida Esperada:



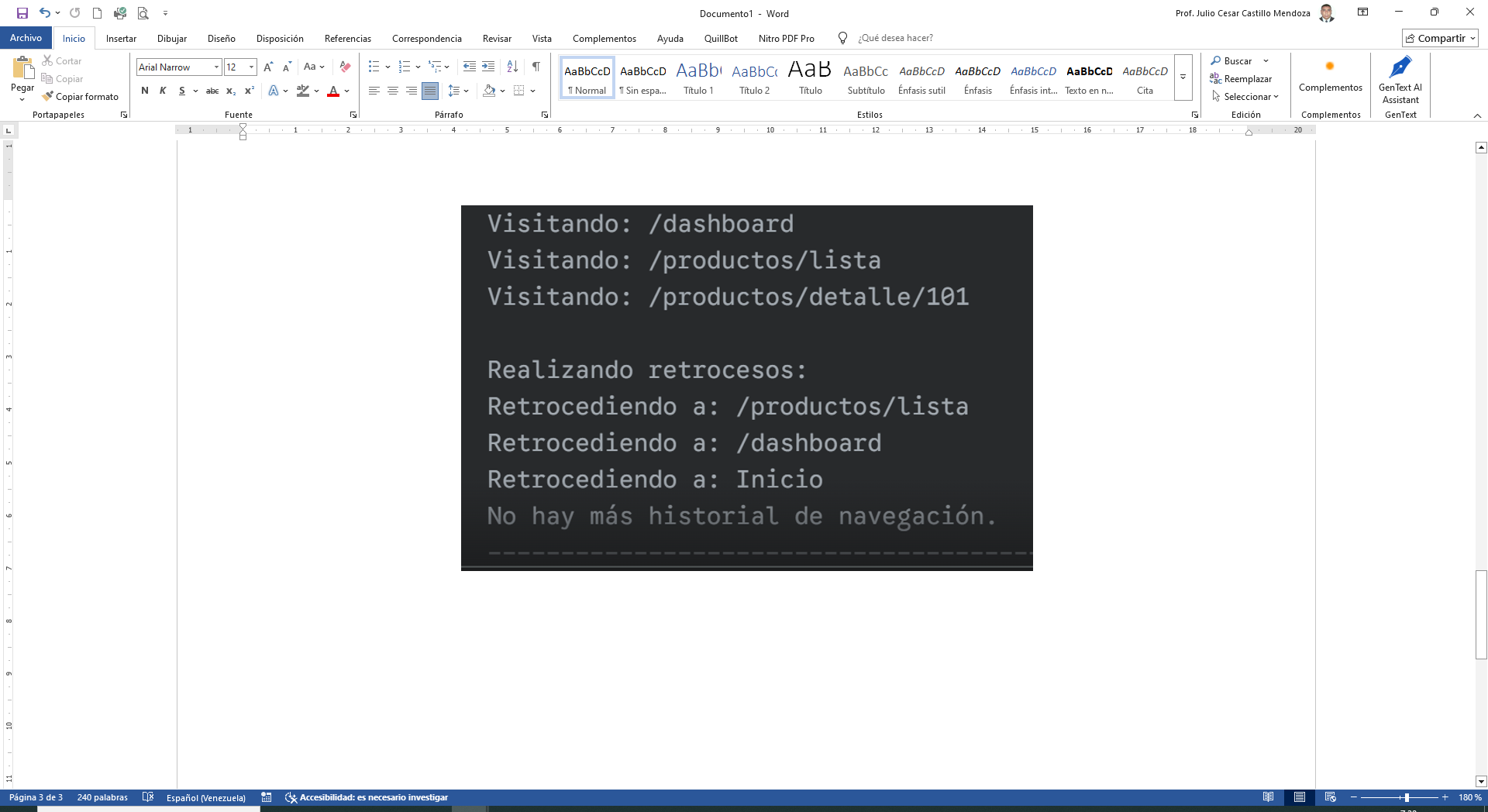
Desafío Práctico:

**Control de Navegación (Páginas Visitadas)**

**Descripción del Problema:** En un sistema de gestión de documentos, CRM o cualquier aplicación con múltiples "vistas" o "páginas", es útil mantener un historial de las páginas visitadas para permitir al usuario "retroceder" a la página anterior de forma sencilla, similar a un navegador web.

**Requisitos Técnicos:**

1. Definir una estructura Pagina que contenga un ID de Página y una URL/Ruta.
2. Utilizar una pila para almacenar las paginasVisitadas.
3. Implementar una función visitarPagina(Pagina): empuja la página actual a la pila.
4. Implementar una función retroceder(): saca la página superior de la pila y simula la navegación a esa página. Devolver la página anterior si existe, o indicar que no hay más historial.
5. Manejar el caso de una pila vacía.

Salida Esperada: