

## Informe

Este informe describe el desarrollo de un programa en C++ para la gestión de estudiantes utilizando listas enlazadas y memoria dinámica. El programa permite agregar estudiantes, ordenar la lista por diferentes criterios, visualizar los datos, y liberar la memoria utilizada al final de la ejecución.

Para gestionar los estudiantes, se ha utilizado una estructura llamada 'Estudiante', la cual contiene el nombre, código y nota del estudiante, además de un puntero a otro nodo de tipo 'Estudiante', lo que permite crear una lista enlazada.

Se implementaron las siguientes funciones principales:

- `agregarEstudiante`: Agrega un nuevo estudiante al inicio de la lista enlazada.
- `mostrarEstudiantes`: Recorre la lista y muestra los datos de cada estudiante.
- `ordenarPorNota`: Ordena la lista de estudiantes de mayor a menor según la nota.
- `ordenarPorCodigo`: Ordena la lista de estudiantes de menor a mayor según el código.
- `liberarMemoria`: Libera la memoria dinámica utilizada por la lista enlazada.

Uno de los principales desafíos fue gestionar correctamente la memoria dinámica para evitar fugas de memoria. Esto se resolvió utilizando la función `liberarMemoria` que recorre la lista enlazada y libera cada nodo antes de finalizar el programa. Otro desafío fue implementar correctamente el algoritmo de ordenamiento para asegurar que los estudiantes se ordenen según los criterios solicitados.

El programa cumple con los requisitos para gestionar estudiantes mediante listas enlazadas, permitiendo un manejo eficiente de la memoria. Las funciones de ordenamiento agregadas permiten organizar a los estudiantes según su nota o su código, lo que facilita la visualización de los datos.