

Anteproyecto: Selector Cíclico de Elementos con Lista Doblemente Enlazada Circular y Replicación Primitiva

Integrantes: Hugo Galina

i. Caso de Uso del Proyecto: Selección Cíclica de Elementos

El proyecto implementará un sistema de selección cíclica de elementos, conceptualmente similar a una pantalla de selección de personajes o ítems en una interfaz de usuario donde la navegación se realiza únicamente en dos direcciones: "siguiente" y "anterior". La característica principal es que, al alcanzar el último elemento y seleccionar "siguiente", se regresa al primer elemento; inversamente, al estar en el primer elemento y seleccionar "anterior", se navega al último.

Cada elemento seleccionable será representado como un nodo. La estructura con sus punteros next y prev, modela de forma natural la navegación bi-direccional. La propiedad "circular" de la lista maneja intrínsecamente el comportamiento de "envolver" del último elemento al primero y viceversa, sin necesidad de lógica condicional adicional compleja para los extremos. Los datos asociados a cada elemento dentro del nodo serán representados por tipos de datos primitivos (str, int, etc.).

ii. Descripción de Operaciones (Funcionalidades) de la Aplicación

La aplicación de consola permitirá al usuario interactuar con el selector cíclico a través de las siguientes operaciones principales, basadas en la manipulación de la CDLL en la versión principal:

1. **Mover al Siguiente Elemento:** Avanza la selección al próximo elemento en el ciclo. En la implementación, esto implica seguir el puntero next del nodo actual.
2. **Mover al Elemento Anterior:** Retrocede la selección al elemento previo en el ciclo. En la implementación, esto implica seguir el puntero prev del nodo actual.
3. **Mostrar Elemento Seleccionado:** Presenta al usuario la información del elemento actualmente marcado (ej: su nombre, ID).
4. **Listar Todos los Elementos:** Muestra la secuencia completa de elementos disponibles en el selector, recorriendo la lista desde el elemento actual o desde un punto fijo.

Adicionalmente, la aplicación gestionará entradas de usuario no válidas para asegurar una ejecución continua y sin errores, cumpliendo así con el requisito de manejo de errores.

iv. Delimitación de Alcance del Proyecto: Replicación Primitiva

Debido a la imposibilidad de gestionar colecciones de forma dinámica o eficiente con solo variables primitivas sueltas, la implementación primitiva solo podrá manejar un número *fijo* y *pequeño* de elementos (8 elementos predefinidos). No se implementarán operaciones de añadir o eliminar elementos en esta versión primitiva. Esta delimitación es fundamental para cumplir con la restricción de solo primitivos y, a su vez, servirá para dimensionar claramente la

utilidad como estructura de datos especializada para resolver este tipo de problemas de forma elegante y escalable, frente a la implementación primitiva que es rígida y limitada.