Operation Flow Explanation

Operation 1: Add Note

What the operation does:

Adds a new note to the user's notepad.

Data Structure used:

Linked List

Why this DS is suitable:

- Notes grow dynamically
- Insertion at the end is fast
- Simple traversal for display

How the user interacts:

- User types a note
- System adds it to the linked list
- User enters a keyword
- System stores the keyword in the hashmap for searching

Operation 2: Undo Action

What the operation does:

Reverses the user's last action.

Data Structure used:

Stack

Why this DS is suitable:

- LIFO structure exactly matches “undo last action”
- Efficient push/pop operations

How the user interacts:

- User selects **Undo**
- The system pops the last action from `undoStack`
- Stores it in `redoStack`

Operation 3: Search Notes

What the operation does:

Searches notes using a keyword.

Data Structure used:

HashMap (unordered_map)

Why this DS is suitable:

- Provides **O(1)** time lookup
- Best structure for fast keyword-based search

How the user interacts:

- User enters a keyword
- System checks if keyword exists in the hashmap
- Displays the matched note