Synopsis: Phone Book Directory Program

Data Structures:

1. **NODE Structure:**
    - Represents an individual contact in the phone book.
    - Contains fields for the contact's number, name, and pointers to the next (**rlink**) and previous (**llink**) contacts.
    - Defined by:

typedef struct Node { int number; char name[30]; struct Node* rlink; struct Node* llink; } NODE;

2. **HEAD Structure:**
    - Represents the head of the linked list of contacts in the phone book.
    - Contains the total number of nodes (**noNodes**) and a pointer to the first contact (**head**).
    - Defined by:

typedef struct head { NODE* head; int noNodes; } HEAD;

3. **Stack:**
    - Maintains the call log to store information about the contacts that have been called.
    - Utilized for displaying call logs and clearing call logs.

Functions:

1. **init():**
    - Initializes the phone book by creating an empty HEAD structure.
    - Returns a pointer to the initialized HEAD structure.
2. **new_node(int no, char name[]):**
    - Creates and returns a new NODE structure with the provided contact number and name.
3. ***insertEnd(HEAD ptr, int no, char name[]):***
    - Inserts a new contact at the end of the phone book.
4. ***insertFront(HEAD ptr, int no, char name[]):***
    - Inserts a new contact at the front of the phone book.

5. ***addContact(HEAD ptr, int no, char name[]):***
    - Adds a new contact to the phone book, handling cases where the name or number already exists.
6. ***deleteFront(HEAD ptr):***
    - Deletes the first contact in the phone book.
7. ***deleteContact(HEAD ptr, int no, char name[]):***
    - Deletes a contact based on either the number or the name.
8. ***displayContacts(HEAD ptr):***
    - Displays all contacts in the phone book.
9. ***searchContact(HEAD ptr, int no, char name[]):***
    - Searches for a contact based on either the number or the name.
10. ***callContact(HEAD ptr, int no, char name[]):***
    - Initiates a call for a contact based on either the number or the name.

11. ***displayCalls(stack<NODE>& calls):***
    - Displays the call log, showing the contacts that have been called.
12. ***clearCallLog(stack<NODE>& calls):***
    - Clears the call log, removing all entries.

The following conclusion can be drawn:

This program provides a functional phone book directory with various features, including adding, deleting, displaying, searching contacts, making calls, and managing call logs. The implementation utilizes linked list data structures (NODE and HEAD) for contact management and a stack for maintaining the call log. The defined functions encapsulate the logic for each operation, contributing to a modular and organized codebase.