22_HEMANT_14.c

```c
/*
Roll no : 22
Batch: A
Author name: Hemant Gupta
Date: 23/08/2024
Description: Program to implement insert,delete,display,search in linked list
*/


#include <stdio.h>
#include <stdlib.h>

// Define the structure for a node
struct Node {
    int data;
    struct Node* next;
};

// Function to insert a node at the end
void insert(struct Node** head_ref, int new_data) {
    struct Node* new_node = (struct Node*)malloc(sizeof(struct Node));
    struct Node* last = *head_ref;
    new_node->data = new_data;
    new_node->next = NULL;

    if (*head_ref == NULL) {
        *head_ref = new_node;
        return;
    }

    while (last->next != NULL)
        last = last->next;

    last->next = new_node;
}

// Function to delete a node by value
void delete(struct Node** head_ref, int key) {
    struct Node* temp = *head_ref;
    struct Node* prev = NULL;

    if (temp != NULL && temp->data == key) {
        *head_ref = temp->next;
        free(temp);
        printf("Entered value is deleted\n");
        return;
    }

    while (temp != NULL && temp->data != key) {
        prev = temp;
        temp = temp->next;
```

```c
    }

    if (temp == NULL) {
        printf("Key not found\n");
        return;
    }

    prev->next = temp->next;
    free(temp);
}

// Function to search for a value in the list
void search(struct Node* head, int key) {
    struct Node* current = head;
    while (current != NULL) {
        if (current->data == key) {
            printf("Found: %d\n", key);
            return;
        }
        current = current->next;
    }
    printf("Not found: %d\n", key);
}

// Function to display the linked list
void display(struct Node* node) {
    while (node != NULL) {
        printf("%d -> ", node->data);
        node = node->next;
    }
    printf("NULL\n");
}

// Driver program to test the above functions
int main() {
    struct Node* head = NULL;
    int choice, value;

    while (1) {
        printf("\n1. Insert\n2. Delete\n3. Display\n4. Search\n5. Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch (choice) {
            case 1:
                printf("Enter value to insert: ");
                scanf("%d", &value);
                insert(&head, value);
                break;

            case 2:
                printf("Enter value to delete: ");
                scanf("%d", &value);
                delete(&head, value);
```

```
                break;

            case 3:
                printf("Linked list: ");
                display(head);
                break;

            case 4:
                printf("Enter value to search: ");
                scanf("%d", &value);
                search(head, value);
                break;

            case 5:
                exit(0);

            default:
                printf("Invalid choice\n");
        }
    }
    return 0;
}

```