include <stdio.h>

#include <stdlib.h>

struct Node {

int data;

struct Node \*next;

};

struct Node \*insertAtBeginning(struct Node \*head, int value) {

struct Node \*newNode = (struct Node \*)malloc(sizeof(struct Node));

newNode->data = value;

newNode->next = head;

return newNode;

}

struct Node \*insertAtEnd(struct Node \*head, int value) {

struct Node \*newNode = (struct Node \*)malloc(sizeof(struct Node));

newNode->data = value;

newNode->next = NULL;

if (head == NULL) {

return newNode;

}

struct Node \*current = head;

while (current->next != NULL) {

current = current->next;

}

current->next = newNode;

return head;

}

struct Node \*deleteNode(struct Node \*head, int value) {

if (head == NULL) {

return NULL;

}

if (head->data == value) {

struct Node \*temp = head;

head = head->next;

free(temp);

return head;

}

struct Node \*current = head;

while (current->next != NULL && current->next->data != value) {

current = current->next;

}

if (current->next != NULL) {

struct Node \*temp = current->next;

current->next = current->next->next;

free(temp);

}

return head;

}

// Function to display the linked list

void display(struct Node \*head) {

struct Node \*current = head;

while (current != NULL) {

printf("%d -> ", current->data);

current = current->next;

}

printf("NULL\n");

}

int main() {

struct Node \*head = NULL;

head = insertAtBeginning(head, 200);

head = insertAtBeginning(head, 10);

head = insertAtEnd(head, 305);

display(head);

head = deleteNode(head, 10);

display(head);

return 0;

}

Output:

10 -> 200 -> 305 -> NULL

200 -> 305 -> NULL

--------------------------------

Process exited after 0.009291 seconds with return value 0

Press any key to continue . . .