```c
#include <inttypes.h>
#include <stdio.h>
#include <stdlib.h>

struct Node {
  int data;
  struct Node* nxp;
};

struct Node* XOR(struct Node* a, struct Node* b)
{
  return (struct Node*)((uintptr_t)(a) ^ (uintptr_t)(b));
}

struct Node* insert(struct Node** head, int value)
{
  if (*head == NULL) {
    struct Node* node
      = (struct Node*)malloc(
        sizeof(struct Node));

    node->data = value;
    node->nxp = XOR(NULL, NULL);
    *head = node;
  }
  else {
    struct Node* curr = *head;
    struct Node* prev = NULL;
    struct Node* node = (struct Node*)malloc(sizeof(struct Node));
    curr->nxp = XOR(node, XOR(NULL, curr->nxp));
    node->nxp = XOR(NULL, curr);
    *head = node;
    node->data = value;
  }
  return *head;
}
void printList(struct Node** head)
  struct Node* curr = *head;
  struct Node* prev = NULL;
  struct Node* next;
  while (curr != NULL) {
    printf("%d ", curr->data);
    next = XOR(prev, curr->nxp);
    prev = curr;
    curr = next;
  }
}

struct Node* delEnd(struct Node** head)
{
  if (*head == NULL)
    printf("List is empty\n");
  else {
    struct Node* curr = *head;
    struct Node* prev = NULL;
    struct Node* next;
    while (XOR(curr->nxp, prev) != NULL) {
      next = XOR(prev, curr->nxp);
      prev = curr;
      curr = next;
    }
    if (prev != NULL)
      prev->nxp = XOR(XOR(prev->nxp, curr), NULL);
    else
      *head = NULL;
    free(curr);
        printf("Deleted Last Element!\n");
  }
  return *head;
}

int main ()
{
    printf("XOR Linked List\n");
    int op,a;
    struct Node *head = NULL;
    for(;;){
        printf("\n1.Insert\n2.Delete\n3.Display\n4.Exit\nEnter your choice:");
        scanf("%d",&op);

        switch(op){
            case 1:
                printf("Enter Element:");
                scanf("%d",&a);
                insert(&head, a);
                printf("Element Inserted!\n");
                break;
            case 2:
                delEnd(&head);
                break;
            case 3:
                if (head == NULL)
                    printf("List is empty\n");
                else{
                    printList(&head);
                    printf("\n");
                }
                break;
            case 4:
                exit(0);
            default:
                printf("Invalid Choice!");
                break;
        }
    }
    return(0);

}
```

```
XOR Linked List

.Insert
2.Delete
3.Display
4.Exit
Enter your choice:1
Enter Element:50
Element Inserted!

.Insert
2.Delete
3.Display
4.Exit
Enter your choice:1
Enter Element:60
Element Inserted!

1.Insert
2.Delete
3.Display
4.Exit
Enter your choice:1
Enter Element:70
Element Inserted!

1.Insert
2.Delete
3.Display
4.Exit
Enter your choice:3
70 60 50

1.Insert
2.Delete
3.Display
4.Exit
Enter your choice:2
Deleted Last Element!

1.Insert
2.Delete
3.Display
4.Exit
Enter your choice:3
```