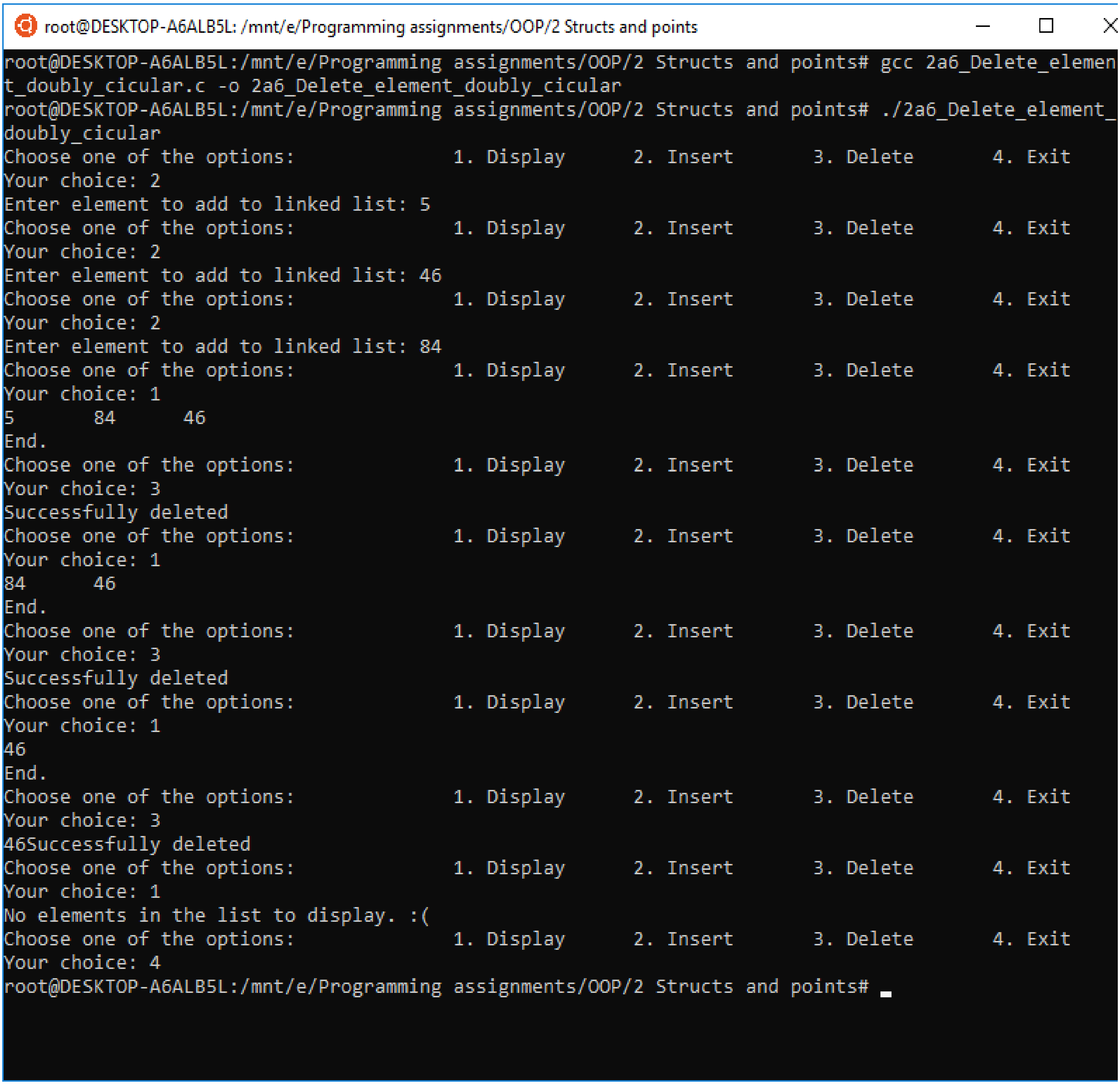
Output :



root@DESKTOP-A6ALB5L: /mnt/e/Programming assignments/OOP/2 Structs and points

```
root@DESKTOP-A6ALB5L:/mnt/e/Programming assignments/OOP/2 Structs and points# gcc 2a6_Delete_elemen
t_doubly_cicular.c -o 2a6_Delete_element_doubly_cicular
root@DESKTOP-A6ALB5L:/mnt/e/Programming assignments/OOP/2 Structs and points# ./2a6_Delete_element_
doubly_cicular
Choose one of the options:              1. Display      2. Insert       3. Delete       4. Exit
Your choice: 2
Enter element to add to linked list: 5
Choose one of the options:              1. Display      2. Insert       3. Delete       4. Exit
Your choice: 2
Enter element to add to linked list: 46
Choose one of the options:              1. Display      2. Insert       3. Delete       4. Exit
Your choice: 2
Enter element to add to linked list: 84
Choose one of the options:              1. Display      2. Insert       3. Delete       4. Exit
Your choice: 1
5       84      46
End.
Choose one of the options:              1. Display      2. Insert       3. Delete       4. Exit
Your choice: 3
Successfully deleted
Choose one of the options:              1. Display      2. Insert       3. Delete       4. Exit
Your choice: 1
84      46
End.
Choose one of the options:              1. Display      2. Insert       3. Delete       4. Exit
Your choice: 3
Successfully deleted
Choose one of the options:              1. Display      2. Insert       3. Delete       4. Exit
Your choice: 1
46
End.
Choose one of the options:              1. Display      2. Insert       3. Delete       4. Exit
Your choice: 3
46Successfully deleted
Choose one of the options:              1. Display      2. Insert       3. Delete       4. Exit
Your choice: 1
No elements in the list to display. :(
Choose one of the options:              1. Display      2. Insert       3. Delete       4. Exit
Your choice: 4
root@DESKTOP-A6ALB5L:/mnt/e/Programming assignments/OOP/2 Structs and points# _
```

```c
struct node* delete(struct node *handle){
    if(handle==NULL){
        printf("No elements to delete :(\n");
        return handle;
    }
    if(handle->next==handle){
        printf("%d",handle->next->data);
        handle = NULL;
        return handle;
    }
    handle->next->prev = handle->prev;
    handle->prev->next = handle->next;
    handle = handle->next;
    return handle;
}
int main(){
    int el, choice;
    while(1){
        printf("Choose one of the options:\t\t");
        printf("1. Display\t");
        printf("2. Insert\t");
        printf("3. Delete\t");
        printf("4. Exit\n");
        printf("Your choice: ");
        scanf("%d",&choice);
        switch (choice){
        case 1:
            display(handle);
            break;
        case 2:
            printf("Enter element to add to linked list: ");
            scanf("%d",&el);
            handle = insert(el,handle);
            break;
        case 3:
            handle = delete(handle);
            printf("Successfully deleted\n");
            break;
        default:
            return 0;
        }
    }   return 0;
}
```

Code :

```
// Write a function to delete an element into Doubly Circular Linked List
#include<stdio.h>
#include<stdlib.h>
int el, pos;

struct node
{
    int data;
    struct node *prev;
    struct node *next;
}*handle, *new, *current;

void display(struct node *handle){
    if(handle==NULL){
    printf("No elements in the list to display. :(\n");
    return;
    }
    current = handle;
    do{
    printf("%d\t",current->data);
    current = current->next;
    }while(current!=handle);
    printf("\nEnd.\n");
}

struct node* insert(int ele,struct node *handle){
    new = (struct node*)malloc(sizeof(struct node));
    new ->data=ele;
    if(handle==NULL){
        printf("null");
        new ->next=new;
        new->prev=new;
        handle = new;
        return handle;
    }
    new->prev=handle;
    new->next=handle->next;
    handle->next->prev=new;
    handle->next=new;
    return handle;
}
```