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

struct node
    {
        int data;
        struct node *next;
    }*head=NULL;

void createLL()
{

    int n,i,item;
    printf("//////////////////////////LINKED LIST//////////////////////////");
    printf("\n Enter the No. of elements in List\n");
    scanf("%d",&n);
    struct node *q=head;

    for(i=1;i<=n;i++)
    {
        struct node *p=(struct node *)malloc(sizeof(struct node));
        printf("\n enter the element");
        scanf ("%d",&item);
        p->data=item;

        if(head==NULL)
        {
            p->next=NULL;
            head=p;
            q=p;
        }
        else
        {
            q->next=p;
            p->next=NULL;
            q=q->next;
        }

    }
}

void checkPalindrome(struct node *h)
{ int flag=0;
    struct node *f=h;struct node *s=h; struct node *n=h;
    while(f!=NULL&&f->next!=NULL)
    {
        f=f->next->next;
        if(f!=NULL)
        {
            s=s->next;
        }
    }
    if(f==NULL)// LL in even size
    {
        printf("LL is even in length/// ");
        struct node *neww = s->next;
        s->next=NULL;
        struct node *q=NULL; struct node *r=neww;
        while(r!=NULL)
        {
                r=r->next;
                neww->next=q;
                q=neww;
                neww =r;
        }
        while(n!=NULL)
```

```c
67              {
68                  if(n->data==q->data)
69                  {
70                      n=n->next;
71                      q=q->next;
72                      flag=1;
73                      // printf(" palindrome\n\n");
74                  }
75
76
77                  else
78                  {
79                      printf("Not a palindrome");
80                      return;
81                  }
82
83              }
84
85          }
86
87
88      else    // LL is odd size
89          {
90          printf("LL is odd in length/// ");
91          struct node *neww = s->next;
92          struct node *q=NULL; struct node *r=neww;
93          while(r!=NULL)
94          {
95                  r=r->next;
96                  neww->next=q;
97                  q=neww;
98                  neww =r;
99          }
100         while(n->next!=s)
101         {
102             if(n->data==q->data)
103             {
104                 n=n->next;
105                 q=q->next;
106                 flag=1;
107                 // printf(" palindrome\n\n");
108             }
109
110
111              else
112              {
113                  printf("Not a palindrome");
114                  return;
115              }
116
117         }
118
119
120         }
121
122 if(flag==1) printf("\n\n LL is Palindrome");
123 }
124
125 void main()
126 {
127     createLL();
128     checkPalindrome(head);
129 }
```