

```

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

struct node {
    int num;
    struct node *nextptr;
} *stnode;

void createNodeList(int n);
void displayList();

int main() {
    int n;

    printf("\n\n Linked List : To create and display Singly Linked List :\n");
    printf("-----\n");

    printf(" Input the number of nodes : ");
    scanf("%d", &n);

    createNodeList(n);

    int c=countList();
    printf("\n Number of nodes : %d\n",c);

    printf("\n Data entered in the list : \n");
    displayList();

    return 0;
}

void createNodeList(int n) {
    struct node *fnNode, *tmp;
    int num, i;

    stnode = (struct node *)malloc(sizeof(struct node));

    if(stnode == NULL) {
        printf(" Memory can not be allocated.");
    } else {

```

```

printf(" Input data for node 1 : ");
scanf("%d", &num);
stnode->num = num;
stnode->nextptr = NULL;
tmp = stnode;

for(i = 2; i <= n; i++) {
    fnNode = (struct node *)malloc(sizeof(struct node));

    if(fnNode == NULL) {

        printf(" Memory can not be allocated.");
        break;
    } else {

        printf(" Input data for node %d : ", i);
        scanf(" %d", &num);

        fnNode->num = num;
        fnNode->nextptr = NULL;

        tmp->nextptr = fnNode;
        tmp = tmp->nextptr;
    }
}
}

int countList()
{
    int ctr = 0;
    struct node *tmp;
    tmp = stnode;

    // Counting the nodes by traversing the linked list
    while(tmp != NULL) {
        ctr++;
        tmp = tmp->nextptr;
    }
    return ctr;
}

void displayList() {

```

```
struct node *tmp;

if(stnode == NULL) {
    printf(" List is empty.");
} else {
    tmp = stnode;

    printf("Linked List :\n");
    while(tmp != NULL) {
        printf("%d-->", tmp->num);
        tmp = tmp->nextptr;
    }
}
printf("NULL");
}
```