

exp_5/05.c

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 struct Node {
5     int data;
6     struct Node* next;
7 };
8
9 void insertAtFront(struct Node** head_ref, int new_data) {
10    struct Node* new_node = (struct Node*) malloc(sizeof(struct Node));
11    new_node->data = new_data;
12    new_node->next = *head_ref;
13    *head_ref = new_node;
14 }
15
16 void insertAtEnd(struct Node** head_ref, int new_data) {
17    struct Node* new_node = (struct Node*) malloc(sizeof(struct Node));
18    struct Node* last = *head_ref;
19    new_node->data = new_data;
20    new_node->next = NULL;
21    if (*head_ref == NULL) {
22        *head_ref = new_node;
23        return;
24    }
25    while (last->next != NULL) {
26        last = last->next;
27    }
28    last->next = new_node;
29 }
30
31 void printList(struct Node* node) {
32    while (node != NULL) {
33        printf("%d -> ", node->data);
34        node = node->next;
35    }
36    printf("NULL\n");
37 }
38
39 int main() {
40    struct Node* head = NULL;
41    int choice, value;
42    while (1) {
43        printf("Enter 1 to insert at front, 2 to insert at end, 3 to display, 0 to
44 exit: ");
45        scanf("%d", &choice);
46        if (choice == 0) break;
47        switch (choice) {
48            case 1:
```

```
48         printf("Enter value to insert at front: ");
49         scanf("%d", &value);
50         insertAtFront(&head, value);
51         break;
52     case 2:
53         printf("Enter value to insert at end: ");
54         scanf("%d", &value);
55         insertAtEnd(&head, value);
56         break;
57     case 3:
58         printList(head);
59         break;
60     default:
61         printf("Invalid choice\n");
62     }
63 }
64 return 0;
65 }
66 }
```