Aim

Insert node at beginning, end, and after a given node.

Algorithm

- At beginning: newNode->next = head.
- At end: traverse to last node, newNode->next=NULL.
- After node: adjust links accordingly.

C Code

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

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

struct Node* createNode(int data){
    struct Node* n=(struct Node*)malloc(sizeof(struct Node));
    n->data=data; n->next=NULL; return n;
}

void printList(struct Node* head){
    while(head){ printf("%d -> ",head->data); head=head->next; }
    printf("NULL\n");
}

void insertBeginning(struct Node** head,int data){
    struct Node* n=createNode(data);
    n->next=*head; *head=n;
}

void insertEnd(struct Node* head,int data){
    struct Node* n=createNode(data);
    while(head->next) head=head->next;
    head->next=n;
}

void insertAfter(struct Node* prev,int data){
    if(!prev) return;
```

```
    struct Node* n=createNode(data);
    n->next=prev->next; prev->next=n;
}

int main(){
    struct Node* head=NULL;
    insertBeginning(&head,10);
    insertBeginning(&head,20);
    insertEnd(head,30);
    insertAfter(head->next,25);
    printList(head);
    return 0;
}
```

Input

Insert: 20 (begin), 10 (begin), 30 (end), 25 (after 20).

Output

```
20 -> 10 -> 25 -> 30 -> NULL
```

Result

Insertion works in all positions.