1.// insertion of the data in the linkedlist

```
#include <stdio.h>
#include <stdlib.h>
struct Node {
 int data;
 struct Node* next;
};
void insertAtBeginning(struct Node** head, int value) {
 struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
 newNode->data = value;
 newNode->next = *head;
 *head = newNode;
}
void insertAtEnd(struct Node** head, int value) {
 struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
 struct Node* temp = *head;
 newNode->data = value;
 newNode->next = NULL;
 if (*head == NULL) {
 *head = newNode;
 return;
 }
 while (temp->next != NULL) {
 temp = temp->next;
 }
 temp->next = newNode;
```

```
}
void insertAtPosition(struct Node** head, int value, int position) {
 if (position <= 0) {
 printf("Invalid position\n");
 return;
 }
 if (position == 1 || *head == NULL) {
 insertAtBeginning(head, value);
 return;
 }
 struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
 newNode->data = value;
 struct Node* temp = *head;
 int count = 1;
 while (count < position - 1 && temp->next != NULL) {
 temp = temp->next;

 count++;
 }
 if (count < position - 1) {
 printf("Invalid position\n");
 return;
 }
 newNode->next = temp->next;
 temp->next = newNode;
}
```

```
void displayLinkedList(struct Node* head) {
 struct Node* temp = head;
 if (temp == NULL) {
 printf("Linked list is empty.\n");
 return;
 }
 while (temp != NULL) {
 printf("%d -> ", temp->data);
 temp = temp->next;
 }
 printf("NULL\n");

}
int main() {
 struct Node* head = NULL;
 insertAtBeginning(&head, 10);
 insertAtBeginning(&head, 20);
 insertAtBeginning(&head, 30);
 printf("Linked list after insertion at the beginning: ");
 displayLinkedList(head);
 insertAtEnd(&head, 40);
 insertAtEnd(&head, 50);
 printf("Linked list after insertion at the end: ");
 displayLinkedList(head);
 insertAtPosition(&head, 25, 2);
 insertAtPosition(&head, 35, 4);
```

```
 printf("Linked list after insertion at specific positions: ");
 displayLinkedList(head);
 return 0;
}
```

output:

Linked list after insertion at the beginning: 30 -> 20 -> 10 -> NULL

Linked list after insertion at the end: 30 -> 20 -> 10 -> 40 -> 50 -> NULL

Linked list after insertion at specific positions: 30 -> 25 -> 20 -> 35 -> 10 -> 40 ->

50 -> NULL