```
#include <stdio.h>
#include <stdlib.h>
struct Node {
   int data;
    struct Node* prev;
    struct Node* next;
};
struct Node* createNode(int data) {
    struct Node* newNode = (struct Node*) malloc(sizeof(struct Node));
    newNode->data = data;
   newNode->prev = NULL;
    newNode->next = NULL;
   return newNode;
void insertAtBeginning(struct Node** head, int data) {
    struct Node* newNode = createNode(data);
    if (*head == NULL) {
        *head = newNode;
        return;
   newNode->next = *head;
    (*head)->prev = newNode;
    *head = newNode;
void insertAtPos(struct Node** head, int data, int pos) {
    struct Node* newNode = createNode(data);
    struct Node* temp = *head;
    int count = 1;
    if (pos <= 1) {
        insertAtBeginning(head, data);
        return;
    while (temp != NULL && count < pos - 1) {
        temp = temp->next;
        count++;
    if (temp == NULL) {
       printf("Position out of range\n");
```

```
return;
    newNode->next = temp->next;
    if (temp->next != NULL)
        temp->next->prev = newNode;
    temp->next = newNode;
    newNode->prev = temp;
void insertAtEnd(struct Node** head, int data) {
    struct Node* newNode = createNode(data);
    if (*head == NULL) {
        *head = newNode;
        return;
    struct Node* temp = *head;
    while (temp->next != NULL) {
        temp = temp->next;
    temp->next = newNode;
    newNode->prev = temp;
void displayList(struct Node* head) {
    if (head == NULL) {
        printf("List is empty.\n");
        return;
    struct Node* temp = head;
    while (temp != NULL) {
        printf("%d ", temp->data);
        temp = temp->next;
    printf("\n");
int main() {
    struct Node* head = NULL;
    int choice, value, position;
   while (1) {
```

```
printf("1. Insert at Beginning \n");
printf("2. Insert at position \n");
printf("3. Insert at End \n");
printf("4. Display list \n");
printf("5. Quit \n");
printf("Enter your choice: ");
scanf("%d", &choice);
switch (choice) {
    case 1:
        printf("Enter value to insert at Beginning: ");
        scanf("%d", &value);
        insertAtBeginning(&head, value);
        break;
    case 2:
        printf("Enter value: ");
        scanf("%d", &value);
        printf("Enter position: ");
        scanf("%d", &position);
        insertAtPos(&head, value, position);
        break;
    case 3:
        printf("Enter value: ");
        scanf("%d", &value);
        insertAtEnd(&head, value);
        break;
    case 4:
        printf("Current Doubly linked list: \n");
        displayList(head);
        break;
    case 5:
        printf("Quitting program.\n");
        exit(0);
    default:
        printf("Invalid choice");
```

## Output:

```
soft-MIEngine-In-mdztszem.y3z --stdout=Microsoft-MIEngin
gz1csgsw.w5t --dbgExe=C:\\msys64\\ucrt64\\bin\\gdb.exe -

    Insert at Beginning

Insert at position
Insert at End
4. Display list
5. Quit
Enter your choice: 1
Enter value to insert at Beginning: 1

    Insert at Beginning

Insert at position
3. Insert at End
4. Display list
5. Quit
Enter your choice: 1
Enter value to insert at Beginning: 2

    Insert at Beginning

2. Insert at position
Insert at End
4. Display list
5. Quit
Enter your choice: 4
Current Doubly linked list:
2 1

    Insert at Beginning

Insert at position
Insert at End
4. Display list
5. Quit
```

```
Enter your choice: 3
Enter value: 4
1. Insert at Beginning
2. Insert at position
3. Insert at End
4. Display list
5. Quit
Enter your choice: 4
Current Doubly linked list:
2 1 4

    Insert at Beginning

2. Insert at position
3. Insert at End
4. Display list
5. Quit
Enter your choice: 2
Enter value: 3
Enter position: 4

    Insert at Beginning

Insert at position
3. Insert at End
4. Display list
5. Quit
Enter your choice: 4
Current Doubly linked list:
2 1 4 3
1. Insert at Beginning
2. Insert at position
3. Insert at End
4. Display list
5. Quit
Enter your choice:
```

	PAGE: DATE:
7	# unclude (stelio u)
	# unclude (stellib.h)
	THE ORDINAL STREET
	shuck wool
	2
	crit clata;
	skeet Node feer
	struct Node* nent;
	3)
	for trul 100, 1 x to a foot a state of the
	Struct Node" create Node (cuit data)
4	struct Node new Noole: Cstruct Node ) malloc (spiof (structum)
	Newhoole -> dadfa = dafa;
4	newhode → prev = NULL;
	new Node - neut = NULY
	Achorn new Mode;
	ed ide state-1: 8
	Solut Edas-t-rela
	void insertAt Biginning (struct Node** head, int data)
1	2
1	shuef Node new Node = create Node (data);
ı	y (* head = NULL)
T	S (Single) Still
Ť	* head = new Noch.
t	Rehan;
t	g window
	newwede -> next = + head:
	(* head) -> prev = newNoole,
	* head = newNode;
	3
+	

	PAGE: DATE:	_
(	void until Afterel (smul Node "head; uit data)	
	shuct Node * new Node = creat Node (data);	_
	if (* head = = NULL)	_
	· ·	_
	* head = newNode;	
	Rehan;	
	3	_
- 1	struct Node* temp = * tread:	
	while ( temp > next !=NUW)	
	4	
	temp = temp - nent	
	Š	
	temp > nont = new Noole;	
	new Node -> prev = temp;	
3		
	ging mass	
U	oid disflay dist (skuct Node * head)	
	fred Creeken saf of 1000 /2	
	of Chead = = NULY	
	<b>{</b>	
	fruit ("hist is employ \n");	
	get a compiging	
	&	
	8 truck Nocle * temp = head;	
	while Ctemp! = NULY	
	1	90
	frent ("1-dr", temp-data);	
	Level - touch 3 would	
	3	
4	freit ("12 4)	
7		

PAGE DATE uit main () shuct nodor head = NULL; aid choice, value, forthon; While (1) frent/("1: Zabert of Beginning in"); freigl ("2. Rushet as position (""); freit ("3 level at End 14"); frints (14. Dusplay dist ("1); print( 18. mit 14"); peint ("tube your choice: "); Seauf (4.1.d" & chair) durtch (of choice) case1. fruit (" Entre value to uniet at Beginning!") scanf (" 1.d", & value); unetABeginning (Thead walle); beat Case 2: prentfl"this value: "); Lay ("I'd", & value); weed fruit (" the position") scarfe", d', & portion). with all of ( & head, value, forhis); break Cape 3: prenty (" Pupe value: "). Scenf (1. d', & waln) instracted ( & head, value); break;