

Add a node to left of a node, delete a node and display a doubly linked list.

```
typedef struct Node{
```

```
    int value;
```

```
    struct Node* next;
```

```
    struct Node* preve;
```

```
}node;
```

```
node* head = NULL;
```

```
void add_beg(int value) // add at beginning
```

```
{
```

```
    node* ptr = (node*) malloc(sizeof(node));
```

```
    ptr->value = value;
```

```
    ptr->preve = NULL;
```

```
    ptr->next = head;
```

```
    if (head != NULL)
```

```
        head->preve = ptr;
```

```
    head = ptr;
```

```
}
```

```
void add_key(int value, int key) // add behind key
```

```
{
```

```
    if (head == NULL)
```

```
    node* temp = head;
```

```
    while (temp != NULL) {
```

```
        if (temp->value == key)
```

```
            break;
```

```
        temp = temp->next;
```

```
}
```

```

if (tmp == NULL) {
    printf("No match");
    return;
}

```

```

if (tmp == head)
{
    add_beg(value);
    return;
}

```

```

node* ptr = (node*) malloc(sizeof(node));
ptr->value = value;
ptr->prev = tmp->prev;
ptr->next = tmp;
(tmp->prev)->next = tmp;
tmp->prev = ptr;
}

```

```

void del_key(int key) {

```

```

node* tmp;

```

```

node* tmp;

```

```

if (head == NULL) {

```

```

    printf("List is empty");
    return;
}

```

```

{

```

```

    node* tmp = head;

```

```

    while (tmp != NULL) {

```

```

        if (tmp->value == key)
            break;

```

```

        tmp = tmp->next;

```

```

    }

```

```
if (tmp == head)
{
    if (head -> next == NULL)
    {
        free(head);
        head = NULL;
        return;
    }
    head = head -> next;
    free(head -> prev);
    head -> prev = NULL;
    return;
}

if (tmp == NULL) {
    printf("No match");
    return;
}

if (tmp -> next == NULL)
{
    tmp -> prev -> next = NULL;
    free(tmp);
    return;
}

tmp -> next -> prev = tmp -> prev;
tmp -> prev -> next = tmp -> next;
free(tmp);
}
```



```
void display()
{
    if (head == NULL) {
        printf("List is empty");
        return;
    }
    node* tmp = head;
    printf("List contains: ");
    while (tmp != NULL) {
        printf("%d ", tmp->value);
        tmp = tmp->next;
    }
}
```