

05/02/24

Double linked list :
① create list
② display list
③ insert at the beginning
④ delete node

include <stdio.h>

include <stdlib.h>

struct node {

 struct node * next;

 int data;

 struct node * prev;

};

struct node * start = NULL;

struct node * create_ll (struct node *);

struct node * display (struct node *)

struct node * insert_beg (struct node *)

struct node * insert_before (struct node *)

struct node * delete_node (struct node *)

struct node * create_ll (struct node * start)

{

```
struct node * newnode, * ptr;
int num;
printf (" To enter -1 to end");
printf (" To enter a value");
while (num != -1)
```

{

```
if (start == NULL)
{
```

```
newnode = (struct node*) malloc (sizeof
(struct node));
newnode -> prev = NULL;
```

```
newnode -> data = NULL;
newnode -> next = NULL;
```

```
start = newnode;
```

3

```
else
```

{

```
ptr = start;
```

```
newnode = (struct node*) malloc (size of  
struct node);
```

```
newnode->data = num;
```

```
while (ptr->next != NULL)
```

```
{
```

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

```
}
```

```
ptr->next = newnode;
```

```
newnode->prev = ptr;
```

```
newnode->next = NULL;
```

```
}
```

```
printf ("In Insert, the data:-"),
```

```
scanf ("%d", &num);
```

```
}
```

```
return start;
```

```
};
```

STRUCT node * display (struct node * start)

{

or C size of

struct node * ptr;

ptr = start;

while (ptr != NULL)

{

printf ("1t %d", ptr->data);

ptr = ptr->next;

{

return start;

{

struct node * insert_beg (struct node * start,

{

struct node * newnode;

int num;

~~printf ("In Enter the data");~~~~scanf ("%d", & num);~~

newnode = (struct node *) malloc (5);

newnode->data = num;

start = newnode;

```
newnode -> next = start;  
newnode -> prev = NULL;  
start = newnode;  
return start;  
};
```

struct node * insert before (struct node*)
{

```
struct node *ptr, *newnode;  
int num, val;  
printf ("In Enter the data");  
scanf ("%d", &num);  
printf ("In Enter value before which");  
scanf ("%d", &val);  
newnode = (struct node *) malloc (sizeof (struct node));  
newnode -> data = num;  
ptr = start;  
while (ptr -> data != val)  
};
```

ptr → curr → next;

3

newnode → next = ptr;

newnode → prev = ptr → prev

ptr → prev → next = newnode;

ptr → prev = newnode;

return start;

3;

struct node * delete_node (struct node *
start)
{

struct node * ptr;

int val

printf ("Enter value to be deleted");

scanf ("%d", &val);

ptr = start;

while (ptr → data != val)

ptr = ptr → next

ptr → prev → next = ptr → next;

ptr → next → prev = ptr → prev;

```
free (ptr);
return start;
}

int main ()
{
    int choice;
    printf ("Menu")
    printf ("1. Create list")
    printf ("2. Insert before");
    printf ("3. Delete");
    printf ("4. display");
    do {
        printf ("Enter your choice");
        scanf ("-d", &choice);
        switch (choice)
        {
            case 1: start = create_list (start);
            break;
```

case 2: start = insert - before (start),
break;

case 3: start = delete - node (start),
break;

case 4: start = display (start);
break

3 while (choice != 5);

3