

LAB-5

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

struct node
{
    int rem;
    char name [100];
    char usn [100];
    struct node *next;
};

struct node *head = NULL;
int counter = 0;

void
insert()
{
    struct node *newnode;
    int s;
    char n[100], u[100];
    printf ("Enter name:");
    scanf ("%s", n);
    printf ("Enter remater?");
    scanf ("%d", &s);
    printf ("Enter usn");
    scanf ("%s", u);

    newnode = (struct node*) malloc (sizeof (struct node));
    newnode->rem = s;
    strcpy (newnode->name, n);
    strcpy (newnode->usn, u);
```

```

if (head == NULL)
    printf (">> First node created \n");
new node->next = head;
head = new node;
counter++;
printf ("Node created \n");
}

```

```

void Insertat (int p)
{
    struct node * newnode;
    int i;
    char n[100], u[100];
    printf ("Enter name");
    scanf ("%s", &n);
    printf ("Enter semester");
    scanf ("%d", &i);
    printf ("Enter URN");
    scanf ("%d", &u);
}

```

```

new node = (struct node*) malloc (sizeof (struct node));
newnode->sem = 5;
strcpy (newnode->name, n);
strcpy (newnode->urn, u);
if (p == 1)
{
    printf ("Node of linked list is inserted in first place \n");
    newnode->next = head;
    head = newnode;
    counter++;
}

```

```
else if (head == null & p > 1)
```

```
{
```

```
printf("currently empty !!!");
```

```
return 0;
```

```
}
```

```
else if (p > count + 1)
```

```
{
```

```
printf
```

```
("is not possible since no of pre exists node  
is less than count + 1");
```

```
return;
```

```
}
```

```
else
```

```
{
```

```
struct node * temp1;
```

```
struct node * temp2;
```

```
int count = 1;
```

```
temp1 = head;
```

```
while (count < p - 1)
```

```
{
```

```
temp1 = temp1 -> next;
```

```
count++;
```

```
}
```

```
temp2 = temp1 -> next;
```

```
temp1 -> next = newnode
```

```
newnode -> next = temp2;
```

```
count++;
```

```
printf("node inserted at %d position in  
linked list\n", p);
```

```
}
```

```
}
```

```
int main()
```

```
{
```

```
    int choice, pos;
```

```
    do
```

```
    {
```

```
        printf("\n");
```

```
        printf("1. Insert node at start of list 2. Insert node  
any where in 3. Insert at end of list.  
4. Display list and Exit(1)");
```

```
        printf("\nEnter your choice: ");
```

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

```
        if (choice == 5)
```

```
            break;
```

```
        switch(choice)
```

```
        {
```

```
            Case 1:
```

```
                Insert();
```

```
                break;
```

```
            Case 2:
```

```
                printf("Enter in which position you want to  
enter your node");
```

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

```
                Insert any(pos);
```

```
                break;
```

```
            Case 3:
```

```
                Insert end();
```

```
                break;
```

Case 4:

```
Display();  
break;
```

default :

```
printf("Invalid\n");  
break;
```

} }

```
while (choice != 5);
```

```
return;
```

}