```
#include <stdlib.h>
#include <string.h>
struct node
{
int sem;
char name[50];
char usn[50];
struct node *next;
};
struct node *head= NULL;
int c=0;
void Insertbegining()
{
struct node *newnode;
int s;
char a[50],b[50];
printf("Enter your name  : ");
scanf("%s",a);
printf("Enter your usn  : ");
scanf("%s",b);
printf("Enter your semester  : ");
scanf("%d",&s);

newnode=(struct node*)malloc(sizeof(struct node));
newnode->sem =s;
strcpy(newnode->name,a);
strcpy(newnode->usn,b);

newnode->next=head;
```

```
head=newnode;
c++;
printf("Node created\n");
}
void Insertany(int p)
{
struct node *newnode;
int s;
char a[30],b[30];
printf("Enter your name  : ");
scanf("%s",a);
printf("Enter your usn  : ");
scanf("%s",b);
printf("Enter your semester  : ");
scanf("%d",&s);

newnode=(struct node*)malloc(sizeof(struct node));
newnode->sem =s;
strcpy(newnode->name,a);
strcpy(newnode->usn,b);
if(p==1)
{
printf("Node of linked list is inserted in the first position\n");
newnode->next=head;
head=newnode;
c++;
}
else if(head==NULL && p>1)
{
printf("the list is empty and node cannot be created\n");
return;
}
else if(p>(c+1))
{
```

```
printf("Not possible since number of nodes existing in the list is insufficient\n"
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;
c++;
printf("Node inserted at %d position in linked list\n",p);
}
}

void Insertend()
{
struct node *newnode;
struct node *temp;
int s;
char n[30],u[30];
printf("Enter your name  : ");
scanf("%s",n);
printf("Enter your semester  : ");
scanf("%d",&s);
printf("Enter your usn  : ");
scanf("%s",u);
```

```
newnode=(struct node*)malloc(sizeof(struct node));
newnode->sem =s;
strcpy(newnode->name,n);
strcpy(newnode->usn,u);
if (head==NULL)
{
newnode->next=NULL;
head=newnode;
printf("first node of linked list created\n");
c++;
}
else
{
temp=head;
while(temp->next!=NULL)
{
temp=temp->next;
}
temp->next=newnode;
newnode->next=NULL;
c++;
printf("Node created\n");
}
}
void display()
{
struct node *ptr;
ptr=head;
int i=1;

if(ptr==NULL)
{
printf("Linked list is empty!\n");
}
else
{
```

```
while(ptr!= NULL)
{
printf("----NODE %d----\n",i);
printf("Name: %s\n",ptr->name);
printf("USN: %s\n",ptr->usn);
printf("Sem: %d\n",ptr->sem);
printf("\n");
i++;
ptr=ptr->next;
}

}

}


int main()
{
int choice,pos;
do
{

printf("\n1. Insert node at beginning of the list\n2. Insert node anywhere in t
printf("\nEnter your choice : ");
scanf("%d",&choice);
if(choice==5)
break;
switch(choice)
{
case 1:
Insertbegining();
break;

case 2:
```

```
printf("Enter in which position of the list you want to enter your node\n");
scanf("%d",&pos);
Insertany(pos);
break;

case 3:
Insertend();
break;

case 4:
display();
break;

default:
printf("Wrong choice!\n");
break;
}
}while(choice!=5);
return 0;
}
```

```
1. Insert node at beginning of the list
2. Insert node anywhere in the list
3. Insert at the end of list
4. Display list
5. Exit

Enter your choice : 1
Enter your name  : ANVI
Enter your usn  : 1BM19CS021
Enter your semester  : 3
Node created

1. Insert node at beginning of the list
2. Insert node anywhere in the list
3. Insert at the end of list
4. Display list
5. Exit

Enter your choice : 3
Enter your name  : AAA
Enter your semester  : 1BM19CS001
Enter your usn  : Node created
```

```
1. Insert node at beginning of the list
2. Insert node anywhere in the list
3. Insert at the end of list
4. Display list
5. Exit

Enter your choice : 4
----NODE 1----
Name: ANVI
USN: 1BM19CS021
Sem: 3

----NODE 2----
Name: AAA
USN: BM19CS001
Sem: 1


1. Insert node at beginning of the list
2. Insert node anywhere in the list
3. Insert at the end of list
4. Display list
5. Exit
```

```
Enter your choice : 2
Enter in which position of the list you want to enter your node
2
Enter your name  : ABC
Enter your usn  : 1BM19CS003
Enter your semester  : 3
Node inserted at 2 position in linked list

1. Insert node at beginning of the list
2. Insert node anywhere in the list
3. Insert at the end of list
4. Display list
5. Exit

Enter your choice : 4
----NODE 1----
Name: ANVI
USN: 1BM19CS021
Sem: 3

----NODE 2----
Name: ABC
USN: 1BM19CS003
Sem: 3
```

```
1. Insert node at beginning of the list
2. Insert node anywhere in the list
3. Insert at the end of list
4. Display list
5. Exit

Enter your choice : 4
----NODE 1----
Name: ANVI
USN: 1BM19CS021
Sem: 3

----NODE 2----
Name: AAA
USN: BM19CS001
Sem: 1


1. Insert node at beginning of the list
2. Insert node anywhere in the list
3. Insert at the end of list
4. Display list
5. Exit
```