Date:

LAB 5

Page No.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
struct node;
{ int sem;
char name[30];
char usn[30];
struct node *next;
}; struct node *head =NULL;
int counter =0;
void Insertbeg()
{ struct node *newnode;
int s;
char n[30], u[30];
printf("Enter the name:");
scanf("%s", n);
printf("Enter the Semester:");
scanf("%d", &s);
printf("Enter the USN:");
scanf("%s", &u);
new node = (struct node*)malloc(size of(struct node));
newnode->sem =s;
strcpy(newnode->name, n);
strcpy(newnode->usn, u);
if (head==NULL)
```

```
printf("First node in linked list is created\n");
newnode->next=head;
head=newnode;
counter++;
printf("Node created\n");
} void Insertany(int p)
{ struct node *newnode;
int S;
char n[30], u[30];
printf("Enter the name: ");
scanf("%s",n);
printf("Enter the semester: ");
scanf("%d",&S);
printf("Enter the USN: ");
scanf("%s",&u);
newnode = (struct node*) malloc(sizeof(struct node));
newnode->sem = S;
strcpy(newnode->name, n);
strcpy(newnode->usn, u);
if(p==1)
{printf("Node is inserted in first position\n");
newnode->next = head;
head=newnode;
counter++; }
else if(head==NULL && p>1)
```

Date:

Page No.

```
{ printf("List is empty\n");
return;
} else if (p>(counter +1))
{ printf("Not possible since number of pre-existing
 nodes in 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;
 counter ++;
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];
```

Date:

Page No.

```
printf("Enter the name: ");
scanf("%s", n);
printf("Enter the semester: ");
scanf("%d", &s);
printf("Enter the 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");
counter++;
} else {
temp = head;
while(temp->next != NULL)
{ temp = temp->next;
} temp->next = newnode;
newnode->next = NULL;
counter++;
printf("Node created.\n");
} }
```

Date:

Page No.

```
void display()
{struct node *ptr;
ptr = head;
int i=1;
if(ptr == NULL){ 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("\n");
printf("\n1. Insert node at beginning \n2. Insert
node at specified position \n3. Insert at the end
of list \n4. Display list \n5. Exit \n");
printf("\n Enter your choice: ");
scanf("%d", &choice);
if(choice==5)
break; switch(choice)
{ case 1: Insertbeg();
break;
```

Date:

Page No.

```
case 2: printf ("Enter node at specified position\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;
}
```