lab 6:

Name: AMIT . R
USN: 1BM19CS016

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
void create ();
void del (char *);
void del_beg ();
void del_end ();
void display ();
void insertbeg ();
struct node
{
    char name[20];
    char id[10];
    int sem;
    struct node * next;
};
struct node * head = NULL;
void main ()
{
    int c; char ele[10];
    do
    {
        printf ("enter choice : 1. create 2. display 3.
            delete 4. insertbefore 5. deletebegning 6. delete
            end 7. exit \n");
        scanf ("%d", &c);
        switch (c)
        {
            case 1: create (); break;
            case 2: display (); break;
            case 3: printf ("Enter elmnt id to be deleted \n");
                scanf ("%s", ele); break;
```

```
            case 5: del_beg(); break;
            case 6: del_end(); break;
            case 7: exit(0); break;
        }
    }while(1);
}
void create()
{
    struct node *newnode, *temp;
    char n[20], id[10]; int s;
    int s;
    newnode = (struct node*)malloc(sizeof(struct node));
    printf("Enter data: name, id & sem\n");
    scanf("%s%s%d", n, id, &s);
    strcpy(newnode->name, n);
    strcpy(newnode->id, id);
    newnode->sem = s;
    if(head==NULL)
    {
        newnode->next = NULL;
        head = newnode;
        printf("Node is created\n");
    }
    else
    {
        temp = head;
        while(temp->next!=NULL)
        { temp = temp->next; }
        temp->next = newnode;
        newnode->next = NULL;
        printf("Node created\n");
```

```
    }
}
void display()
{
    struct node *ptr = NULL;
    ptr = head;
    if (ptr == NULL)
     printf("No element to print\n");
    else
    {
        while (ptr != NULL)
        {
            puts (ptr -> name);
            puts (ptr -> id);
            printf ("%d\n", ptr -> sem);
            printf ("....\n");
            ptr = ptr -> next;
        }
    }
}
void del (char id1[10])
{
    struct node *temp, *del = NULL;
     if (head == NULL)
      printf ("Empty list\n");
     temp = head;
     if (strcmp (head -> id, id1) == 0)
     {
            head = head -> next;
            exit(0);
     }
     while (temp -> next != NULL)
```

```
{
    if (strcmp (temp->next->id, id) == 0)
    {
        del = temp->next;
        if (del->next == NULL)
        temp->next = NULL;
        else
         temp->next = del->next;
    }
    else
     temp = temp->next;
}
void insertbeg()
{
    struct node *newnode;
    char n[20], id1[10]; int s;
    printf ("Enter name, sem, id at beg");
    scanf ("%s %s %d", n, id1, &s);
    newnode = (struct node*) malloc (sizeof (struct node));
    strcpy (newnode->name, n);
    strcpy (newnode->id, id);
    scanf ("%d", &s);
    newnode->next = head;
    head = newnode;
}
void del_beg()
{
    struct node * ptr;
    if (head == NULL)
    printf ("Empty list\n");
    else
    {
        ptr = head;
        head = ptr->next;
```

```
        free (ptr);
        printf ("begning node deleted \n");
    }
}
void del_end()
{
    struct node *temp, *ptr;
    if (head == NULL)
        printf ("Empty list \n");
    else if (head -> next == NULL)
    {
        head = NULL;
        free (head);
        printf ("Single node deleted\n");
    }
    else
    {
        ptr = head;
        while (ptr -> next != NULL)
        {
            temp = ptr;
            ptr = ptr -> next;
        }
        temp -> next = NULL;
        free (ptr);
        printf ("Deleted node from last.n\n");
    }
}
```