Page No.
Date

NAME: Anshul H. Surana
USN: 1BM19CS020

LAB-6

Q.6) WAP to Implement Singly link list with following operations.
a.) Create a linked list b.) Deletion of first element, specified element and last element in the list.
c.) Display the contents of the linked list.

```
→ #include <stdio.h>
#include <stdlib.h>
struct node
{
        int id ;
      char name [20];
      int sem;
      struct node* next;
};
struct node* head = NULL;

void linkedList();
void insertNode();
void deleteNode(int);
void deleteNodeAtBegin();
void deletenodeOfGiven();
void displayList();

int size=0;
int main()
{ linkedlist();
      return 0;
}
```

Anshul H. Surana

Scanned by TapScanner

Page No.
Date

NAME: ANSHUL H. SURANA
USN: 1BM19CS020

```
void linkedlist()
{ int choice 1, choice 2;
  printf("\n Enter the Operation");
  printf("\n Insert Node \n 2.-Delete node \n
     3. Display list \n 4. Exit. \n choice");
  scanf("%d", &choice 1);
  switch (choice 1)
{ Case 1 : insertNode();
           break;
  case 2: printf("\n 1. At the First Position
  \n 2. At End of list \t 3. At Given
  Element Position \n Choice: ");
  scanf( "%d", &choice 2);
  switch (choice 2)
{ case 1: deleteNode(1);
          break;
  Case 2: deleteNode(2);
          break;
  case 3: deleteNode(3);
          break;
  default: Printf("\n Input Error, try Again \n");
        linkedlist();
  } break;
  Case 3: displaylist();
        break;
  case 4: exit(0);
  default: printf("\n Input error, Try again! \n");
        linkedlist();
    }
}
```

Anshul H. Surana

Page No.
Date

NAME: ANSHUL H. SURANA
USN: 1BM19CS020

```
void insertnode()
{ struct node *newnode *temp;
  newnode =(struct node*) malloc(sizeof(struct node));
  printf("\n Enter the Details");
  ~~printf~~("\n ID:");          scanf("%d", &(newnode->id));
  printf("Name:");           scanf("%s", (newnode->name));
  printf("Sem:");            scanf("%d", &(newnode->sem));

 if (head==NULL)
 { newnode ->next =NULL;
     head= newnode;
     printf ("\n Node Created);
     linkedlist();
     size++;
 }
for(temp=head; (temp->next)!=NULL; temp=(temp->next));
  newnode-> next=NULL;
 {
       temp-> next= newnode;
        size++;
       printf("\n Node Created");
         linkedlist();
   }
}
void   deleteNode (int flag)
{   if (head==NULL)
     {
   printf("\n The list is Empty\n");
    linkedlist();
 }
else
 if(head->next==NULL)
```

Anshul H. Surana

NAME: ANSHUL H. SURANA
USN: 1BM19CS020

Page No
Date

```
  { printf("\n Node Deleted \n Now List is empty");
    free(head);
     head = NULL;
     linkedlist();
  }
  else
  { switch (flag)
    { case 1: deleteNodeAtBegin();
              break;
      case 2: deleteNodeAtEnd();
              break;
      case 3: deleteNode of Given();
              break;
    }
    linkedlist();
  }
}
void deleteNodeAtBegin()
{ struct node* temp=head;
  head= head→next;
  free(temp);
  printf("\n Node Deleted");
}
void deleteNodeAtEnd()
{ struct node* temp= head;
   head= head→next;
   free(temp→next); temp→next=NULL;
   printf("\n Node Deleted");
}
void deleteNodeOfGiven()
{ struct node* temp1=head, *temp2=temp1;
```

Anshul H. Surana

Page No.
Date

NAME: ANSHUL H. SURANA
USN: 1BM19CS020

```
int ele;
printf("\n Enter the element");
scanf("%d", &ele);
for( temp1; temp1->next != NULL ; temp1 = temp1->
{       temp2 = temp1 ->next;
        if( temp2 -> id = = ele )
        {  temp1 ->next = temp2 ->next;
           free(temp2);
          printf( "\n  Node  Deleted ");
          return;
     }
    if (temp1->  id = = ele)
    {      head = temp2;
          free (temp1);
          printf("\n Node Deleted");
          return;
      }
    }
   printf("\n Element Not present in the List");
    linkedList();
}
void displayList()
{  if (head = = NULL)
   {  Printf("\n Empty List \n");
       linkedList();      }
     printf( "\n The List is:");
  for (struct node * temp=head; temp != NULL; temp=temp
 { printf ("\n Student Details");
  printf ("\n ID: %d ", temp->id);
  printf ("\n Name : %s ", temp->name);
  printf("\n Sem: %d ", temp->sem);
 }  ~~[illegible]~~      linked List();
}
```

Anshul H. Surana