LAB-5

NAME: ANSHUL H. SURANA
USN: 1BM19CS026

Q5) WAP to Implement Singly Link list with following operations
a) Create a linked list b) Insertion of a node at first position, at any position and at the end of 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 insertNodeAtBegn();
void insertNodeAtEnd();
void insertNode AtAny();
void  display List();

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

Anshul H. Surana

NAME: ANSHUL H. SURANA
USN: 1BM19CS020

```
void  linkedList()
{  int  choice1, choice2;
   printf("\n  Enter the operation");
   printf("\n 1. Insert Node     \n 2. Display List \n 3.
               \nChoice : ");
   scanf("%d", &choice1);
   switch (choice1)
 { case1: printf ("\n 1. At the First Position\t
               2. At End of List \t 3. At any Spec
    Location\n  Choice:");
    scanf(" %d", &choice2);
    switch  (choice2)
   {  case1:  insertNodeAtBegin();
              break;
      case 2:  insertNodeAtEnd();
              break;
      case 3:  insertNodeAtAny();
              break;
     default: printf ("\n Input Error, try Again!\n")
              linkedlist();
   } break;
  case 2:   displayList();
            break;
 case 3:  exit(0);
 default: printf("\n Input error, Try again!\n");
          linkedlist ();
  }
}
void   insertNodeAtBegin()
{ struct node* newnode.
  newnode = (struct node*) malloc (size of (struct node))
```

Anshul H.Sura

```
    printf("\n Enter the Details");
    printf ("\n ID:");      scanf("%d", &(newnode→id));
    printf("Name:");        scanf("%s", &(newnode→name));
    printf("Sem:");         scanf("%d", &(newnode→sem));

    newnode → next=head;
    head= newnode;
     size++;
   printf("\n  Node  created \n");
    linkedlist();
}
void   insertNodeAtEnd()
{ struct   node *  newnode   * temp;
  newnode  =(struct node*) malloc (size of (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("Node   Created \n");
   linkedlist();
}
or (temp= head;  (temp→next) != NULL; temp= (temp→
                                              next));
newnode →  next = NULL;
  temp→next = newnode;
     size ++;
  printf("\n Node Created);
     linkedlist();
}
```

Anshul H. Surana

Page No.
Date

NAME: ANSHUL H. SURANA
USN: 1BM19CS020

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

int pos=0, s=0;
 printf("\n Enter the position (pos>=1 and pos<%d)",
                                                  size);

 scanf("%d", &pos);
  if(pos==0)
{  printf("\n Error position, check the operation menu!");
     linkedlist();
}
for(temp; temp->next!=NULL; temp= temp->next)
{  if(s==(pos-1))
  { newnode->next= (temp->next);
    temp->next = newnode;
    size++;
    printf("\n   Node   Created\n");
     linkedlist();
  }
   s++;
  temp= temp->next;
  }
}
void displaylist()
{ if(head=NULL)
 { printf("\n Empty List\n");
   linkedlist();
}
```

Anshul H.Surana

NAME: ANSHUL H. SURANA
USN: 1BM19CS020

```
printf("\n   The List is:");
 for(struct node * temp = head; temp != NULL;
                     temp=temp->next)
{ printf("\n  Student  Details ");
 printf(" \nID : %d ", temp->id);
 printf("\n Name: %s", temp->name);
 printf("\n Sem:  %d", temp->sem);
}
 linkedlist();
}
```

Anshul H. Surana