```
struct node
{
  int data;
  struct node *next;
};
struct node *head = NULL;
int main()
{ int choice, ele;
  char ch;
  int n=1;
  do
  {
   printf("\n1.Create \n2. Display \n3. Delete");
   printf("\n Enter your choice:");
   scanf("%d", &choice);
   switch(choice)
   {
     case 1: create();
             break;
     case 2: display();
             break;
     case 3: printf("Enter the element to be
              deleted\n");
             scanf("%d", &ele);
             delfun(ele);
             break;
   }
  }while(n==1);
}
void create()
{
```

Disha B
1BM19CS050

```
struct node * newnode, *temp;
int item;
newnode = (struct node*) malloc(sizeof(structnode))
printf("Enter the data");
scanf("%d",&item);
newnode->data=item;
if(head==NULL)
{
  newnode->next=NULL;
  head=newnode;
  printf("Node 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("Nothing to print\n");
  }
else
  {
```

```
while(ptr!=NULL)
  {
    printf("%d",ptr->data);
    ptr= ptr->next;
  }
 }
}
void delfun(int ele)
 { struct node *temp, *del = NULL;
   if(head== NULL)
    { printf("Empty list Cant delete\n");
      return;
    }
   temp= head;
   if (head->data==ele)
    { head = head->next;
      return;
    }
   while(temp->next!= NULL)
   { if(temp->next->data ==ele)
      { del=temp->next;
        if (del->next== NULL)
            temp->next=NULL;
        else
          temp->next=del->next;
      }
     else
      temp = temp->next;
   }
  if(del==NULL)
  { printf "element not found in the list\n");
  }} return;
```