Ajith.MS
1BM19CS010

LAB-7

```
#include <stdlib.h>
#include<string.h>
 struct node
 {
   int rem;
 struct node *next;
 };

 struct node *head=NULL;
 struct node *head2=NULL;
   int c=0;
  void main()
   {
        struct node *newnode;
        struct node *temp;
 int s;
  printf("enter item:");
  scanf("%d",&s);
 newnode=(struct node*)malloc(sizeof(struct node));
   newnode->rem=s;
   if (head=NULL)
   {
      newnode->next=NULL
      head =newnode;
      printf("first node of cc created)
        c++;
      }
```

```
else
{
  temp = head;
  while (temp->next != NULL)
  {
    temp = temp->next;
  }
  temp->next = newnode;
  newnode->next = NULL;
  ct++;
  printf("Node created\n");
}
}

void insert2()
{
  struct node *newnode;
  struct node *temp;
  int n, y;
  printf("Enter element to create list2\n");
  do
  {
    printf("Enter integer : \n");
    scanf("%d", &n);
    newnode = (struct node*)malloc(sizeof(node));
    newnode->data = n;
    if (head2 == NULL)
    {
      newnode->next = NULL;
      head2 = newnode;
      printf("First node of Linked list created\n");
      ct++;
    }
    else
```

```
{
  temp = head;
  while(temp->next != NULL)
{
  temp = temp->next;
}
temp->next = newnode;
  newnode->next = NULL;
    ctr;
  printf("node created");
printf("do you want to continue adding: 0 or 1");
    scanf("%d", &y);
  } while(y != 0);
}

void bubble sort()
{ int swapped, i;
  struct node* ptr1;
  struct node *lptr = NULL;
if (head == NULL)
    return;

do
{ swapped = 0;
  ptr1 = head;
  while (ptr1->next != lptr)
{ if (ptr1->data > ptr1->next->data)
{ int temp = ptr1->data;
  ptr1->data = ptr1->next->data;
  ptr1->next->data = temp;
```

```
        swapped=1;
      }
      ptr1=ptr1->next;
    }
    ptr=ptr1;
  }
  while(swapped);
}
void reverse()
{
  struct node* prev=NULL;
  struct node* curr=head;
  struct node* next=NULL;
  while (curr!=NULL)
  {
    next=curr->next;
    curr->next=prev;
    prev=curr;
    curr=next;
  }
head=prev;
}

void concat()
{
  struct node *ptr;
  if(head==NULL)
  {
    head=head2;
  }
  if(head2==NULL)
  {
    head2=head;
```

```
while(ptr->next != NULL)
    ptr = ptr->next;
    ptr->next = head2;
}

int main()
{
  int choice, n;
  do
  {
    printf("1. Insert node 2. Sort node 3. Reverse node,
        4 - Concat 2 lists 5. exit");
    scanf("%d", &choice);
  switch (choice)
    {
    case 1: insert();
        break;
    case 2: bubblesort();
        break;
    case 3: reverse();
        break;
    case 4: insert2();
        concat();
        break;
    case 5: break
    default: printf("wrong choice\n");
        break
    } while(choice != 5);
      return 0;
  }
```