Week 6

**1. (Concatenating Lists) write a program that concatenates two linked lists of characters. The program should include function concatenate that takes pointers to both lists as arguments and concatenates the second list to the first list.**

**Code:**

#include <stdio.h>

#include <stdlib.h>

struct node

{

int data;

struct node \*next;

};

display(struct node \*head)

{

if(head == NULL)

{

printf("NULL\n");

}

else

{

printf("%d\n", head -> data);

display(head->next);

}

}

int concatenate(struct node \*a,struct node \*b)

{

if( a != NULL && b!= NULL )

{

if (a->next == NULL)

a->next = b;

else

concatenate(a->next,b);

}

else

{

printf("Either a or b is NULL\n");

}

}

int main()

{

struct node \*prev,\*a, \*b, \*p;

int n,i;

printf ("number of elements in a:");

scanf("%d",&n);

a=NULL;

for(i=0;i<n;i++)

{

p=malloc(sizeof(struct node));

scanf("%d",&p->data);

p->next=NULL;

if(a==NULL)

a=p;

else

prev->next=p;

prev=p;

}

printf ("number of elements in b:");

scanf("%d",&n);

b=NULL;

for(i=0;i<n;i++)

{

p=malloc(sizeof(struct node));

scanf("%d",&p->data);

p->next=NULL;

if(b==NULL)

b=p;

else

prev->next=p;

prev=p;

}

int concats = concatenate(a,b);

printf("Concatenated result is %d \n",concats);

}

**Output:**

