/\*link list concatination\*/

#include<stdio.h>

#include<stdlib.h>

struct node

{

int info;

struct node \*link;

};

struct node \*create\_list(struct node \*);

struct node \*concat( struct node \*start1,struct node \*start2);

struct node \*addatbeg(struct node \*start, int data);

struct node \*addatend(struct node \*start,int data);

void display(struct node \*start);

int main()

{

struct node \*start1=NULL,\*start2=NULL;

start1=create\_list(start1);

start2=create\_list(start2);

printf("First list is : ");

display(start1);

printf("Second list is : ");

display(start2);

start1=concat(start1, start2);

printf("Concatenated list is : ");

display(start1);

}/\*End of main()\*/

struct node \*concat( struct node \*start1,struct node \*start2)

{

struct node \*ptr;

if(start1==NULL)

{

start1=start2;

return start1;

}

if(start2==NULL)

return start1;

ptr=start1;

while(ptr->link!=NULL)

ptr=ptr->link;

ptr->link=start2;

return start1;

}

struct node \*create\_list(struct node \*start)

{

int i,n,data;

printf("Enter the number of nodes : ");

scanf("%d",&n);

start=NULL;

if(n==0)

return start;

printf("Enter the element to be inserted : ");

scanf("%d",&data);

start=addatbeg(start,data);

for(i=2;i<=n;i++)

{

printf("Enter the element to be inserted : ");

scanf("%d",&data);

start=addatend(start,data);

}

return start;

}/\*End of create\_list()\*/

void display(struct node \*start)

{

struct node \*p;

if(start==NULL)

{

printf("List is empty\n");

return;

}

p=start;

while(p!=NULL)

{

printf("%d ", p->info);

p=p->link;

}

printf("\n");

}/\*End of display() \*/

struct node \*addatbeg(struct node \*start,int data)

{

struct node \*tmp;

tmp=(struct node \*)malloc(sizeof(struct node));

tmp->info=data;

tmp->link=start;

start=tmp;

return start;

}/\*End of addatbeg()\*/

struct node \*addatend(struct node \*start, int data)

{

struct node \*p,\*tmp;

tmp= (struct node \*)malloc(sizeof(struct node));

tmp->info=data;

p=start;

while(p->link!=NULL)

p=p->link;

p->link=tmp;

tmp->link=NULL;

return start;

}/\*End of addatend()\*/

