Questions

Answers

1.

#include<stdio.h>

struct node{

int data;

struct node \*next;

};

int main(){

struct node \*head,\*ptr;

struct node n1,n2,n3;

head = &n1;

n1.data = 10;

n1.next = &n2;

n2.data = 20;

n2.next = &n3;

n3.data = 30;

n3.next = NULL;

ptr = head;

while(ptr != NULL){

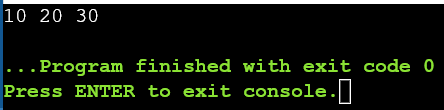
printf("%d ",ptr->data);

ptr = ptr->next;

}

return 0;

}

OUTPUT

2.

#include <stdio.h>

#include <stdlib.h>

struct Node

{

int data;

struct Node \*next;

};

int main (void){

struct Node \*Head;

struct Node \*New;

struct Node \*ptr;

char choice;

Head=NULL;

New=(struct Node\*)malloc(sizeof(struct Node));

printf("Enter a value: ");

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

New->next=NULL;

Head=New;

ptr=Head;

printf("\nDo you want to add another Node?\n\n(Y/N): ");

fflush(stdin);

scanf("%c",&choice);

while(choice=='Y' || choice =='y'){

New=(struct Node\*)malloc(sizeof(struct Node));

printf("Enter a value: ");

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

New->next=NULL;

ptr->next=New;

ptr=ptr->next;

printf("\nDo you want to add another Node?\n\n(Y/N): ");

fflush(stdin);

scanf("%c",&choice);

}

if(choice=='N' || choice=='n')

printf("\nThe linked list contains \n");

ptr=Head;

while (ptr!=NULL)

{

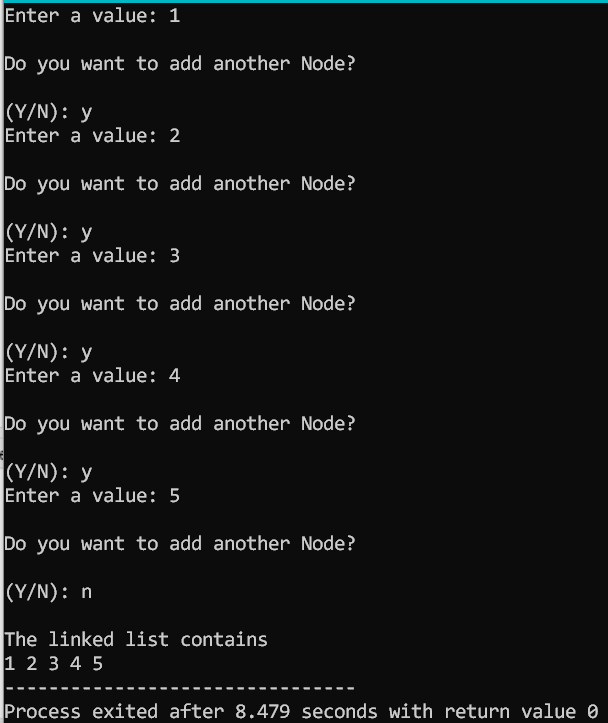
printf("%d ",ptr->data);

ptr=ptr->next;

}

return 0;

}

OUTPUT

3.

#include <stdio.h>

#include <stdlib.h>

struct Node

{

int data;

struct Node \*next;

};

int main (void){

struct Node \*Head;

struct Node \*New;

struct Node \*ptr;

char choice;

Head=NULL;

New=(struct Node\*)malloc(sizeof(struct Node));

printf("Enter a value: ");

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

fflush(stdin);

New->next=NULL;

Head=New;

ptr=Head;

printf("\nDo you want to add another Node?\n\n(Y/N): ");

scanf("%c",&choice);

while(choice=='Y' || choice =='y'){

New=(struct Node\*)malloc(sizeof(struct Node));

printf("Enter a value: ");

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

fflush(stdin);

New->next=NULL;

ptr->next=New;

ptr=ptr->next;

printf("\nDo you want to add another Node?\n\n(Y/N): ");

scanf("%c",&choice);

}

ptr=Head;

int count=0;

while (ptr!=NULL)

{

printf("%d ",ptr->data);

ptr=(ptr->next)->next;

}

ptr=Head;

while(ptr!=NULL){

count++;

ptr=ptr->next;

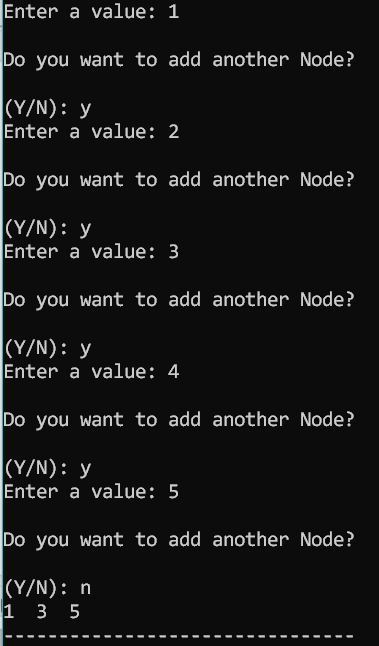
}

printf("\nNumber of Nodes: %d",count);

return 0;

}

OUTPUT



4.

#include <stdio.h>

#include <stdlib.h>

struct Node{

int data;

struct Node\* next;

};

void display(struct Node\* ptr){

while(ptr != NULL){

printf("%d ", ptr->data);

ptr = ptr->next;

}

}

struct Node\* insertbegin(struct Node\* HEAD){

int NEW;

printf ("Enter the element to insert in the begining: ");

scanf ("%d", &NEW);

struct Node\* ptr = (struct Node\*)malloc(sizeof(struct Node));

ptr->next = HEAD;

ptr->data = NEW;

return ptr;

}

struct Node\* insertAtEnd(struct Node\* HEAD){

int NEW;

printf("Enter the element to insert at the end: ");

scanf ("%d", &NEW);

struct Node\* ptr = (struct Node\*)malloc(sizeof(struct Node));

struct Node\* p = HEAD;

while (p->next!=NULL)

{

p = p->next;

}

ptr->data = NEW;

p->next = ptr;

ptr->next = NULL;

return HEAD;

}

struct Node\* insertAtPos(struct Node\* HEAD){

int pos, NEW;

printf ("Enter the element to insert: ");

scanf ("%d", &NEW);

printf ("Enter in which position: ");

scanf ("%d", &pos);

struct Node\* ptr = (struct Node\*)malloc(sizeof(struct Node));

struct Node\* p = HEAD;

int i = 0;

while (i != pos-2)

{

p = p->next;

i++;

}

ptr->data = NEW;

ptr->next = p->next;

p->next = ptr;

return HEAD;

}

int main()

{

struct Node \*HEAD;

struct Node \*NEW;

struct Node \*PTR;

char choice;

int choice1;

HEAD = NULL;

NEW = (struct Node \*)malloc(sizeof(struct Node));

printf("\nEnter data: ");

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

fflush(stdin);

NEW->next = NULL;

HEAD = NEW;

PTR = HEAD;

printf("Do you want to add another node?(Y/N): ");

fflush(stdin);

scanf("%c", &choice);

while (choice == 'Y' || choice == 'y')

{

NEW = (struct Node \*)malloc(sizeof(struct Node));

printf("\nEnter data: ");

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

fflush(stdin);

NEW->next = NULL;

PTR->next = NEW;

PTR = PTR->next;

printf("Do you want to add another node?(Y/N): ");

fflush(stdin);

scanf("%c", &choice);

}

PTR = HEAD;

printf("\nPrinting the List: \n");

while (PTR != NULL)

{

printf("%d ", PTR->data);

PTR = PTR->next;

}

printf("\n\n");

printf ("1. Insert in the beginning\n");

printf ("2. Insert at the end\n");

printf ("3. Insert at a specific position\n");

printf ("\nEnter your choice: ");

fflush(stdin);

scanf ("%d", &choice1);

switch (choice1)

{

case 1:

HEAD = insertbegin(HEAD);

break;

case 2:

HEAD = insertAtEnd(HEAD);

break;

case 3:

HEAD = insertAtPos(HEAD);

break;

default:

printf("\nError: Invalid input\n");

exit(0);

break;

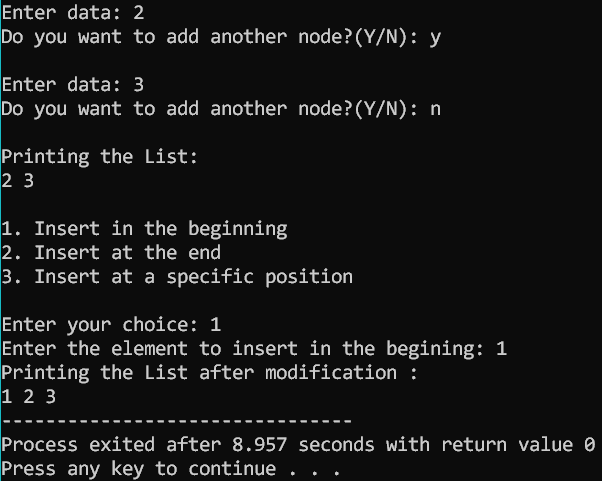
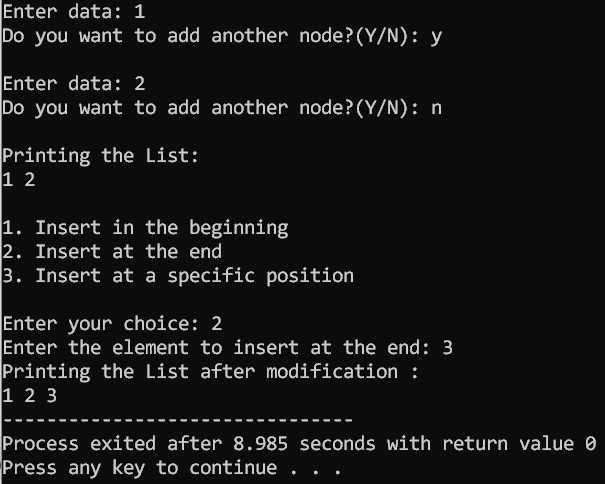
}

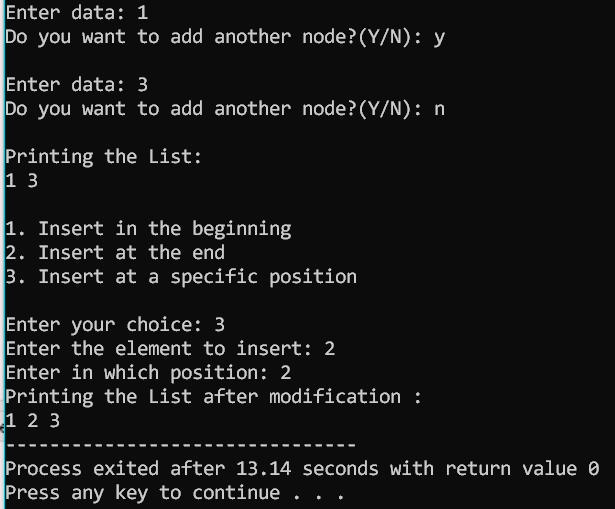
printf("Printing the List after modification : \n");

display(HEAD);

}

OUTPUT





5.

#include <stdio.h>

#include <stdlib.h>

struct Node{

int data;

struct Node\* next;

};

void display(struct Node\* ptr){

while(ptr != NULL){

printf("%d ", ptr->data);

ptr = ptr->next;

}

}

struct Node\* deletestart(struct Node \*HEAD)

{

struct Node \*p = HEAD;

p = p->next;

free(HEAD);

return p;

}

struct Node\* deleteAtPos(struct Node\* HEAD){

int pos;

printf("Enter the position: ");

scanf("%d", &pos);

struct Node\* ptr = HEAD;

int i=0;

while(i!=pos-2){

ptr = ptr->next;

i++;

}

struct Node\* p = ptr->next;

ptr->next = p->next;

free(p);

return HEAD;

}

struct Node\* deleteEnd(struct Node\* head){

struct Node\* ptr = head;

struct Node\* temp;

ptr = head;

while ((ptr->next)->next != NULL)

{

ptr = ptr->next;

}

temp = ptr->next;

ptr->next = NULL;

free(temp);

return head;

}

int main()

{

struct Node \*HEAD;

struct Node \*NEW;

struct Node \*PTR;

char choice;

int choice1;

HEAD = NULL;

NEW = (struct Node \*)malloc(sizeof(struct Node));

printf("\nEnter data: ");

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

fflush(stdin);

NEW->next = NULL;

HEAD = NEW;

PTR = HEAD;

printf("Do you want to add another node?(Y/N): ");

fflush(stdin);

scanf("%c", &choice);

while (choice == 'Y' || choice == 'y')

{

NEW = (struct Node \*)malloc(sizeof(struct Node));

printf("\nEnter data: ");

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

fflush(stdin);

NEW->next = NULL;

PTR->next = NEW;

PTR = PTR->next;

printf("Do you want to add another node?(Y/N): ");

fflush(stdin);

scanf("%c", &choice);

}

PTR = HEAD;

printf("\nPrinting the List: \n");

while (PTR != NULL)

{

printf("%d ", PTR->data);

PTR = PTR->next;

}

printf("\n\n");

printf ("1. Delete the first element\n");

printf ("2. Delete the last element\n");

printf ("3. Delete the node at a specific position\n");

printf ("\nEnter your choice: ");

scanf ("%d", &choice1);

switch (choice1)

{

case 1:

HEAD = deletestart(HEAD);

break;

case 2:

HEAD = deleteEnd(HEAD);

break;

case 3:

HEAD = deleteAtPos(HEAD);

break;

default:

printf("\nError: Invalid input\n");

exit(0);

break;

}

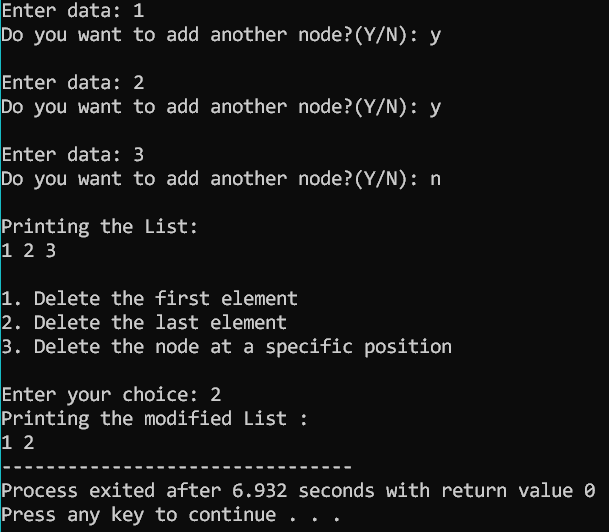
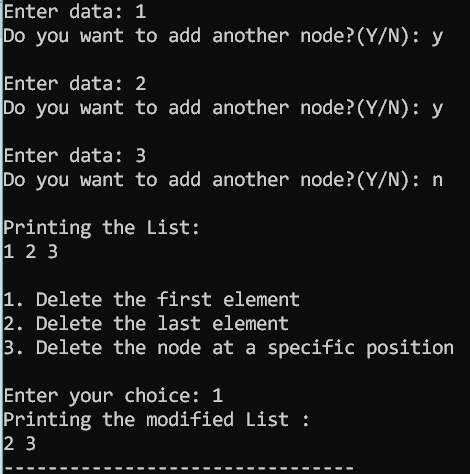
printf("Printing the modified List : \n");

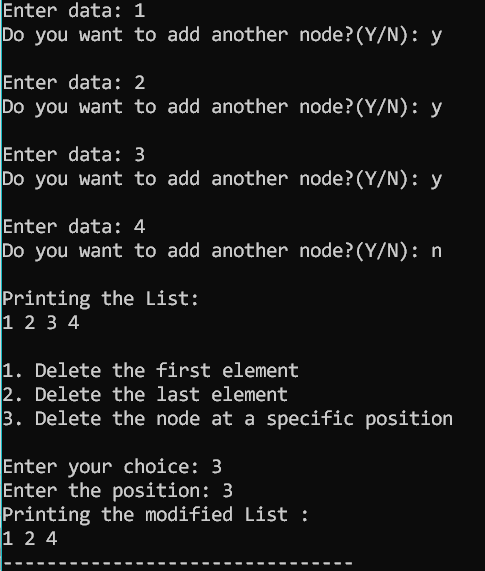
display(HEAD);

return 0;

}

OUTPUT





6.

#include <stdio.h>

#include <stdlib.h>

struct Node

{

int data;

struct Node \*next;

};

int main (void){

struct Node \*Head;

struct Node \*New;

struct Node \*ptr;

char choice;

Head=NULL;

New=(struct Node\*)malloc(sizeof(struct Node));

printf("Enter a value: ");

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

fflush(stdin);

New->next=NULL;

Head=New;

ptr=Head;

printf("\nDo you want to add another Node?\n\n(Y/N): ");

scanf("%c",&choice);

while(choice=='Y' || choice =='y'){

New=(struct Node\*)malloc(sizeof(struct Node));

printf("Enter a value: ");

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

fflush(stdin);

New->next=NULL;

ptr->next=New;

ptr=ptr->next;

printf("\nDo you want to add another Node?\n\n(Y/N): ");

scanf("%c",&choice);

}

ptr->next=Head;

ptr=Head;

printf("\nShowing circular linked list\n");

while(ptr->next!=Head){

printf("%d ",ptr->data);

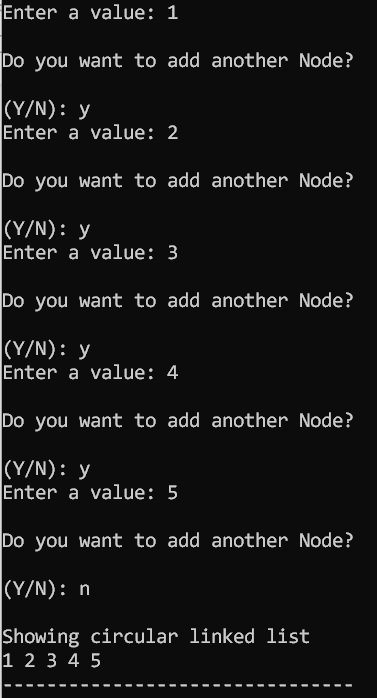
ptr=ptr->next;

}

printf("%d ",ptr->data);

return 0;

}

OUTPUT

7.

#include <stdio.h>

#include <stdlib.h>

struct Node

{

int data;

struct Node \*Link;

};

int main (void){

struct Node \*Head;

struct Node \*New;

struct Node \*PTR;

Head=NULL;

New=(struct Node\*)malloc(sizeof(struct Node));

printf("Enter the value: ");

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

fflush(stdin);

New->Link=NULL;

Head=New;

PTR=Head;

char choice;

printf("Do you want to create another node\n\n(Y/N): ");

scanf("%c",&choice);

fflush(stdin);

while(choice=='Y' || choice=='y') {

New=(struct Node\*)malloc(sizeof(struct Node));

printf("Enter the value: ");

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

fflush(stdin);

New->Link=NULL;

PTR->Link=New;

PTR=PTR->Link;

printf("Do you want to create another node\n\n(Y/N): ");

scanf("%c",&choice);

fflush(stdin);

}

int count=0;

PTR=Head;

while(PTR!=NULL){

count++;

PTR=PTR->Link;

}

printf("The Number of Nodes: %d",count);

int key,flag=0;

printf("\nEnter the number you want to search : ");

scanf("%d",&key);

PTR=Head;

while(PTR!=NULL){

if(PTR->data==key)

flag=1;

PTR=PTR->Link;

}

if (flag==1)

printf("\nSearch Successfull! ");

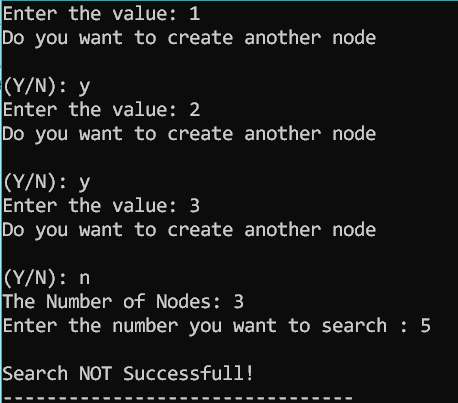
else

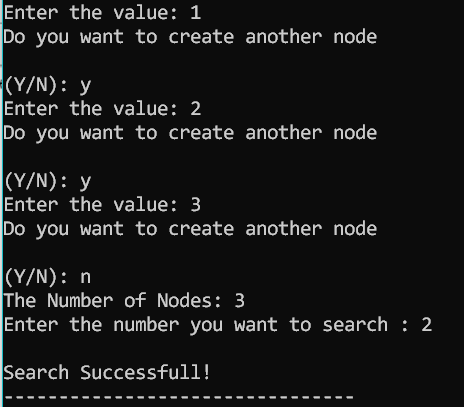
printf("\nSearch NOT Successfull! ");

return 0;

}

OUTPUT





END