#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

void iBeg();

void iEnd();

void ipos();

void delBeg();

void delEnd();

void delpos();

void display();

void search();

int ch, ch2, value, location;

struct Node

{

int data;

struct Node \*previous, \*next;

}\*head = NULL;

void main()

{

do

{

printf("\n1.Insert in begining\n2.Insert at last\n3.Insert at any random location\n4.Delete from Beginning\n 5.Delete from last\n6.Delete the node after the given data\n7.Search\n8.Show\n9.Exit\n");

scanf("%d",&ch);

switch(ch)

{

case 1:

iBeg();

break;

case 2:

iEnd();

break;

case 3:

ipos();

break;

case 4:

delBeg();

break;

case 5:

delEnd();

break;

case 6:

delpos();

break;

case 7:

search();

break;

case 8:

display();

break;

case 9:

break;

default:

printf("Please enter valid choice..");

}

}while(ch<9);

}

void iBeg()

{

printf("\nEnter Item value");

scanf("%d",&value);

struct Node \*newNode;

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

newNode -> data = value;

newNode -> previous = NULL;

if(head == NULL)

{

newNode -> next = NULL;

head = newNode;

}

else

{

newNode -> next = head;

head = newNode;

}

}

void iEnd()

{

printf("\nEnter value");

scanf("%d",&value);

struct Node \*newNode;

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

newNode -> data = value;

newNode -> next = NULL;

if(head == NULL)

{

newNode -> previous = NULL;

head = newNode;

}

else

{

struct Node \*temp = head;

while(temp -> next != NULL)

{

temp = temp -> next;

}

temp -> next = newNode;

newNode -> previous = temp;

}

}

void ipos()

{

printf("\nEnter value");

scanf("%d",&value);

printf("Enter the location");

scanf("%d",&location);

struct Node \*newNode;

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

newNode -> data = value;

if(head == NULL)

{

newNode -> previous =NULL;

newNode -> next = NULL;

head = newNode;

}

else

{

struct Node \*temp1,\*temp2;

temp1=head;

while(temp1 -> data != location&&temp1->next!=NULL)

{

if(temp1 -> next == NULL)

{

printf("Given node is not found !!!");

}

else

{

temp1 = temp1 -> next;

}

}

temp2 = temp1 -> next;

temp1 -> next = newNode;

newNode -> previous = temp1;

newNode -> next = temp2;

temp2 -> previous = newNode;

}

}

void delBeg()

{

if(head == NULL)

printf("List is Empty");

else

{

struct Node \*temp ;

temp= head;

if(temp -> previous == temp -> next)

{

head = NULL;

free(temp);

}

else{

head = temp -> next;

head -> previous = NULL;

free(temp);

}

}

}

void delEnd()

{

struct Node \*ptr;

if(head == NULL)

{

printf("\n UNDERFLOW");

}

else if(head->next == NULL)

{

head = NULL;

free(head);

printf("\nnode deleted\n");

}

else

{

ptr = head;

if(ptr->next != NULL)

{

ptr = ptr -> next;

}

ptr -> previous -> next = NULL;

free(ptr);

printf("\nnode deleted\n");

}

}

void delpos()

{

struct Node \*ptr, \*temp;

int val;

printf("\n Enter the data after which the node is to be deleted : ");

scanf("%d", &val);

ptr = head;

while(ptr -> data != val)

ptr = ptr -> next;

if(ptr -> next == NULL)

{

printf("\nCan't delete\n");

}

else if(ptr -> next -> next == NULL)

{

ptr ->next = NULL;

}

else

{

temp = ptr -> next;

ptr -> next = temp -> next;

temp -> next -> previous = ptr;

free(temp);

printf("\nnode deleted\n");

}

}

void search()

{

struct Node \*ptr;

int item,i=0,flag;

ptr = head;

if(ptr == NULL)

{

printf("\nEmpty List\n");

}

else

{

printf("\nEnter item which you want to search?\n");

scanf("%d",&value);

while (ptr!=NULL)

{

if(ptr -> data == value)

{

printf("\nitem found at location %d ",i+1);

flag=0;

break;

}

else

{

flag=1;

}

i++;

ptr = ptr -> next;

}

if(flag==1)

{

printf("\nItem not found\n");

}

}

}

void display()

{

struct Node \*ptr;

printf("\n printing values...\n");

ptr = head;

while(ptr != NULL)

{

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

ptr=ptr->next;

}

}







