#include <iostream>

#include <stdio.h>

#include <conio.h>

#include <stdlib.h>

using namespace std;

struct node

{

int data;

struct node \*next;

};

struct node \*head;

void insertion\_begin();

void insertion\_sl();

void insertion\_end();

void deletion\_begin();

void deletion\_sl();

void deletion\_end();

void display();

void search();

int main()

{

int choice;

char c;

do

{

cout << "\t\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n";

cout << "\n\t\t\t\t------Menu-----\n";

cout << "\n\t\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n";

cout << "\n\t1.Insertion at beginning";

cout << "\n\t2.Insertion at specific location";

cout << "\n\t3.Insertion at ending";

cout << "\n\t4.Deletion at beginning";

cout << "\n\t5.Deletion at specific location";

cout << "\n\t6.Deletion at ending";

cout << "\n\t7.Display";

cout << "\n\t8.Searching";

cout << "\nEnter your choice: ";

cin >> choice;

switch (choice)

{

case 1:

insertion\_begin();

break;

case 2:

insertion\_sl();

break;

case 3:

insertion\_end();

break;

case 4:

deletion\_begin();

break;

case 5:

deletion\_sl();

break;

case 6:

deletion\_end();

break;

case 7:

display();

break;

case 8:

search();

break;

default:

cout << "\nEnter a valid choice!";

}

cout << "\nDo you want to continue?(Y/N): ";

cin >> c;

} while (c == 'Y' || c == 'y');

return 0;

}

void insertion\_begin()

{

int item;

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

cout << "\nEnter an element:";

cin >> item;

ptr->data = item;

if (head==NULL)

{

head=ptr;

head->next = NULL;

}

else

{

ptr->next = head;

head = ptr;

}

}

void insertion\_sl()

{

int item,loc,i;

struct node \*ptr,\*temp;

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

cout << "\nEnter an element:";

cin >> item;

ptr->data = item;

temp=head;

if (head == NULL)

{

head=ptr;

head->next = NULL;

}

else

{

cout << "\nEnter after which loc to be inserted:";

cin >> loc;

ptr->data=item;

for(i=1;i<=loc;i++)

{

temp=temp->next;

if(temp==NULL)

{

return;

}

}

ptr->next=temp->next;

temp->next=ptr;

}

}

void insertion\_end()

{

int item;

struct node \*ptr,\*temp;

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

cout << "\nEnter an element:";

cin >> item;

ptr->data = item;

temp=head;

if (head == NULL)

{

head=ptr;

head->next = NULL;

}

else

{

while(temp->next!=NULL)

{

temp=temp->next;

}

temp->next=ptr;

ptr->next=NULL;

}

}

void deletion\_begin()

{

struct node \*ptr;

if(head->next==NULL)

{

ptr=head;

head=NULL;

free(ptr);

}

else

{

ptr=head;

head=ptr->next;

free(ptr);

}

}

void deletion\_sl()

{

struct node \*ptr,\*ptr1;

int loc;

cout << "\nEnter the location at which to be deleted:";

cin >> loc;

if(head->next==NULL)

{

ptr=head;

head=NULL;

free(ptr);

}

else

{

ptr=head;

for(int i=1;i<=loc;i++)

{

ptr1=ptr;

ptr=ptr->next;

if(ptr=NULL)

return;

}

ptr1->next=ptr->next;

free(ptr);

}

}

void deletion\_end()

{

struct node \*ptr,\*ptr1;

if(head->next==NULL)

{

ptr=head;

head=NULL;

free(ptr);

}

else

{

ptr=head;

while(ptr->next!=NULL)

{

ptr1=ptr;

ptr=ptr->next;

}

ptr1->next=NULL;

free(ptr);

}

}

void display()

{

struct node \*ptr;

ptr=head;

if(head == NULL)

{

cout << "\nNothing to print";

}

else

{

cout << "\nThe values in list are: \n";

while(ptr!=NULL)

{

cout << ptr->data <<"\n";

ptr=ptr->next;

}

}

}

void search()

{

struct node \*ptr;

int item,i=1;

cout << "\nEnter an item to be searched:";

cin >> item;

ptr=head;

while(ptr!=NULL)

{

if(ptr->data==item)

{

cout << "\n" << item << " is found at position: " << i;

break;

}

else

cout << "\n" << item << "doesn't exist.";

ptr=ptr->next;

i++;

}

}