#include <iostream>

using namespace std;

class Node

{

public:

int data;

Node \*next;

Node(int d)

{

data=d;

next=NULL;

}

};

void insertathead(Node \*&head,int data)

{

Node \*n=new Node(data);

n->next=head;

head=n;

}

int search(Node \*&head,int data)

{

Node \*temp=head;

int z=0;

while(temp!=NULL)

{

z++;

if(temp->data==data)

return z;

temp=temp->next;

}

return -1;

}

//resurssive approach

bool searchrecurssive(Node \*head,int key)

{

if(head==NULL)

return false;

if(head->data==key)

return true;

else

searchrecurssive(head->next,key);

}

void print(Node \*head)

{

while(head!=NULL)

{

cout<<head->data<<" ";

head=head->next;

}

}

int main()

{

Node \*head=NULL;

insertathead(head,10);

insertathead(head,9);

insertathead(head,8);

insertathead(head,6);

insertathead(head,5);

insertathead(head,4);

print(head);

cout<<endl;

cout<<search(head,10);

cout<<endl;

cout<<searchrecurssive(head,5); //return 1 if element is present in linked list

}