#include<iostream>

using namespace std;

struct Nodo{

int valor;

Nodo \*izq,\*der;

};

Nodo \*raiz=NULL;

int main(int argc, char \*\* argv){

Nodo \*aux=raiz;

raiz=new Nodo();

raiz->valor=11;

raiz->izq=new Nodo();

raiz->der=new Nodo();

aux=raiz->izq;

aux->valor=10;

aux->izq=new Nodo();

aux->der=new Nodo();

aux=aux->izq;

aux->valor=5;

aux->izq=NULL;

aux->der=NULL;

aux=raiz->izq;

aux=aux->der;

aux->valor=9;

aux->izq=NULL;

aux->der=NULL;

aux=raiz->der;

aux->valor=8;

aux->izq=NULL;

aux->der=new Nodo();

aux=aux->der;

aux->valor=1;

aux->izq=NULL;

aux->der=NULL;

cout<<raiz->valor<<endl;

cout<<raiz->izq->valor<<" "<<raiz->der->valor<<endl;

cout<<raiz->izq->izq->valor<<" "<<raiz->izq->der->valor<<" "<<raiz->der->der->valor;

}



#include <iostream>

#include <conio.h>

#include <locale.h>

using namespace std;

struct Arbol{

int valor;

Arbol \*izq, \*dere;

};

//Prototipos de funciones

Arbol \*newnodo(int);

void insertar(Arbol \*&,int);

void mostrar(Arbol \*,int);

void InOrden(Arbol \*);

Arbol \*rama=NULL;

int main(){

setlocale(LC\_ALL, "");

int opc,nod,cont=0,val;

do{

cout<<"\t ELIJA ALGUNA OPCION DEL MENÚ\n";

cout<<"\t1.- Insertar Nodos al Arból \n";

cout<<"\t2.- Mostrar Arból \n";

cout<<"\t3.- Recorrido En InOrden \n";

cout<<"\t4.- Salir \n";

cin>>opc;

switch(opc){

case 1:

cout<<"Cuantos nodos desea insertar: \n";

cin>>nod;

for(int i=1; i<=nod; i++){

cout<<"Ingrese el valor del nodo "<<i<<"\n";

cin>>val;

insertar(rama,val);

}

cout<<"\n";

break;

case 2:

cout<<" EL ARBOL ES: \n";

mostrar(rama,cont);

cout<<"\n";

system ("pause");

break;

case 3:

cout<<" EL RECORRIDO EN IN ORDEN ES : \n";

InOrden(rama);

cout<<"\n";

system("pause");

break;

case 4:

cout<<"\*\*\* GRACIAS \*\*\* \n";

break;

}

system("cls");

}while(opc!=4);

return 0;

}

//creando nuevo nodo

Arbol \*newnodo(int num){

Arbol \*nuevo\_nodo=new Arbol();

nuevo\_nodo->valor=num;

nuevo\_nodo->dere=NULL;

nuevo\_nodo->izq=NULL;

return nuevo\_nodo;

}

void insertar(Arbol \*&rama,int num){

if(rama==NULL){

Arbol \*new\_nodo= newnodo(num);

rama=new\_nodo;

}

else{

int raiz=rama->valor;

if(num<raiz){

insertar(rama->izq,num);//insertar datos a la izquierda

}

else{

insertar(rama->dere,num);//insertar datos a la derecha

}

}

}

//mostrar arbol completo

void mostrar(Arbol \*rama, int cont){

if(rama==NULL){ //si arbol vacio

return;

}

else{

mostrar(rama->dere,cont+1);

for(int i=0;i<cont; i++){

cout<<" ";

}

cout<<rama->valor<<endl;

mostrar(rama->izq,cont+1);

}

}

void InOrden(Arbol \*rama){

if(rama==NULL){// si arbol vacio

return;

}

else{

InOrden(rama->izq);

cout<<rama->valor<<" - ";

InOrden(rama->dere);

}

}



