TALLER

1. Genere el árbol binario de búsqueda para la siguiente secuencia de números: 8, 9, 11, 15, 19, 20, 21, 7, 3, 2, 1,5, 6, 4, 13, 14, 10, 12, 17, 16, 18. Analice y describa lo que sucede durante su inserción, genere sus recorridos en preorden, inorden y postorden

Estructura Árbol

#include <iostream>

#include <cstdlib>

using namespace std;

struct nodo{

int nro;

struct nodo \*izq, \*der;

};

typedef struct nodo \*ABB;

ABB crearNodo(int x)

{

ABB nuevoNodo = new(struct nodo);

nuevoNodo->nro = x;

nuevoNodo->izq = NULL;

nuevoNodo->der = NULL;

return nuevoNodo;

}

void insertar(ABB &arbol, int x)

{

if(arbol==NULL)

{

arbol = crearNodo(x);

}

else if(x < arbol->nro)

insertar(arbol->izq, x);

else if(x > arbol->nro)

insertar(arbol->der, x);

}

void preOrden(ABB arbol)

{

if(arbol!=NULL)

{

cout << arbol->nro <<" ";

preOrden(arbol->izq);

preOrden(arbol->der);

}

}

void enOrden(ABB arbol)

{

if(arbol!=NULL)

{

enOrden(arbol->izq);

cout << arbol->nro << " ";

enOrden(arbol->der);

}

}

void postOrden(ABB arbol)

{

if(arbol!=NULL)

{

postOrden(arbol->izq);

postOrden(arbol->der);

cout << arbol->nro << " ";

}

}

void verArbol(ABB arbol, int n)

{

if(arbol==NULL)

return;

verArbol(arbol->der, n+1);

for(int i=0; i<n; i++)

cout<<" ";

cout<< arbol->nro <<endl;

verArbol(arbol->izq, n+1);

}

int alturaAB(ABB arbol)

{

int AltIzq, AltDer;

if(arbol==NULL)

return -1;

else

{

AltIzq = alturaAB(arbol->izq);

AltDer = alturaAB(arbol->der);

if(AltIzq>AltDer)

return AltIzq+1;

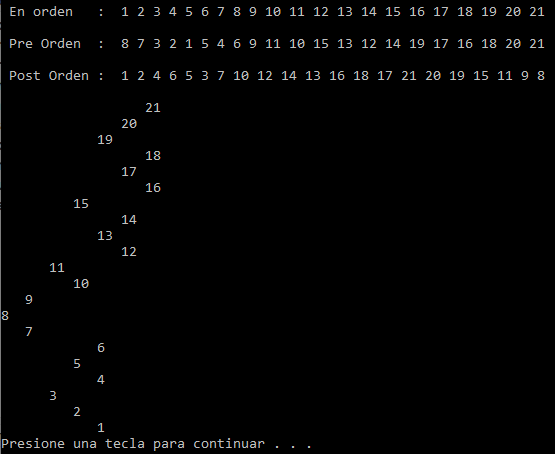
else

return AltDer+1;

}

}

Punto 1:



1. Dada la lista de números: 4, 10, 19, 7, 49, 100, 20, 22, 12
2. Construir el árbol binario de búsqueda, siendo 4 el nodo raíz

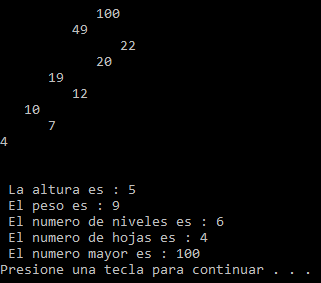
b) Calcular su altura

c) Calcular su peso

d) Calcular el número de niveles

e) Contar el número de hojas

f) retorne el mayor elemento del árbol.



1. Construir un ABB con las claves: 50,25,75,10,40,60,90,35,45,70,42.

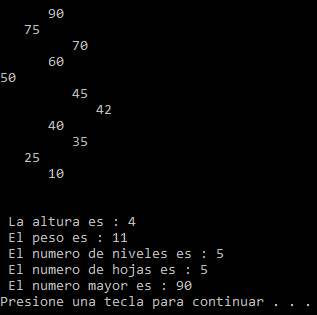
a) Calcular su altura

b) Calcular su peso

c) Calcular el número de niveles

d) Contar el número de hojas

e) retorne el mayor elemento del árbol.



1. Construir un ABB con las claves:10,75,34,22,64,53,41,5,25,74,20,15,90.

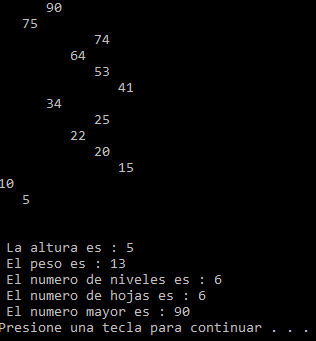
a) Calcular su altura

b) Calcular su peso

c) Calcular el número de niveles

d) Contar el número de hojas

e) retorne el mayor elemento del árbol.

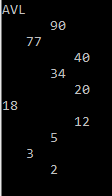


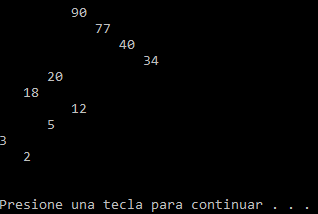
1. Se tiene un árbol AVL vacío que se le insertan, en orden, los siguientes elementos:3, 2, 18, 5, 20, 90, 77, 40, 34, 12

a. Dibuje la disposición final del árbol AVL e indique el número de rotaciones que fueron realizadas.

b. Dibuje la disposición final de un ABB, al que se le inserta esta misma secuencia de números.

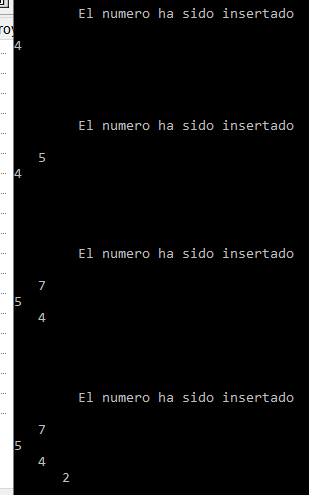
c. Justifique la eficiencia en este tipo de casos, de un AVL sobre un ABB. Use de referencia las disposiciones de las preguntas A y B.

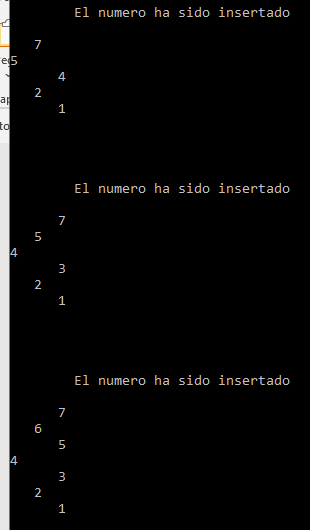
AVL

ABB

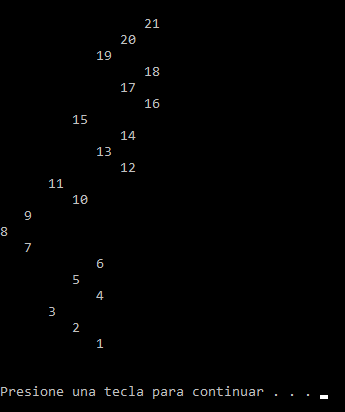
Los árboles AVL son más eficientes que los ABB respecto al balanceo, pues el AVL al estar balanceado hace que el acceso a un nodo sea más rápido, por ejemplo al acceder al nodo 34 en el árbol ABB tendría que realizar los saltos máximos para llegar a él, y en el AVL solo tiene que realizar 2 saltos

1. Dada la siguiente secuencia de números: 4, 5, 7, 2, 1, 3, 6, generar un árbol AVL. Se deberá ilustrar paso a paso el proceso de inserción y rebalanceo.



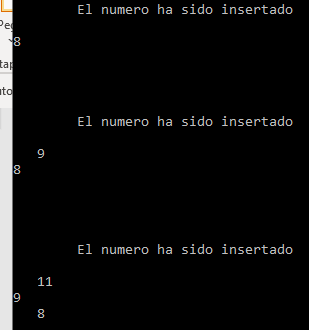


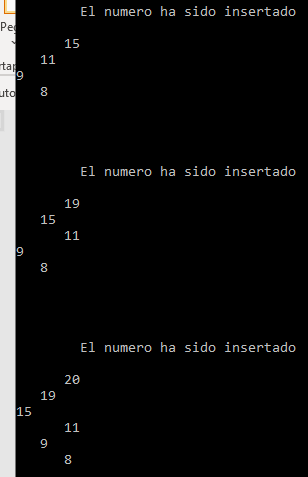
1. Dada la siguiente secuencia de números: 8, 9, 11,15, 19, 20, 21, 7, 3, 2, 1,5, 6, 4, 13, 14, 10, 12, 17, 16, 18, generar:
2. Su árbol binario de búsqueda

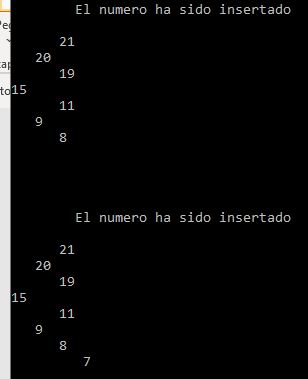


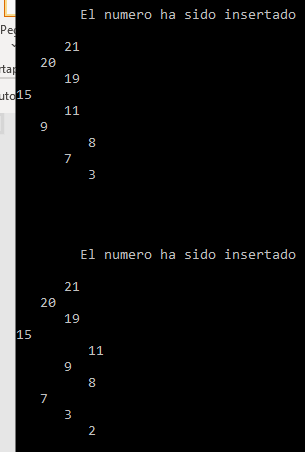
b. Su árbol AVLS e deberá ilustrar paso a paso en cada uno de ellos, el

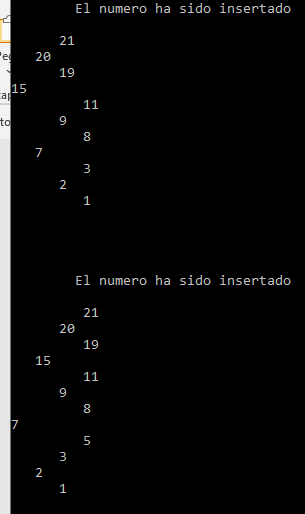
proceso de inserción.

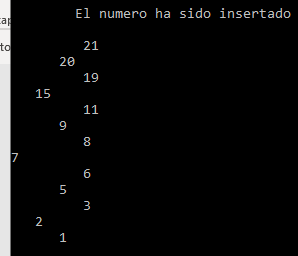


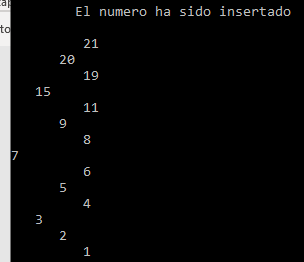


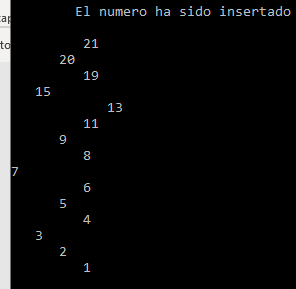


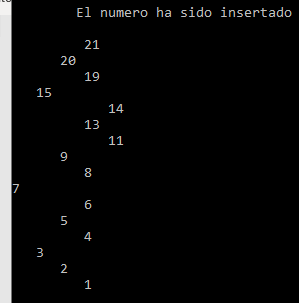


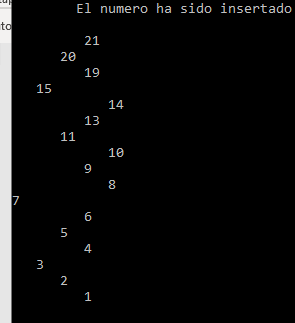


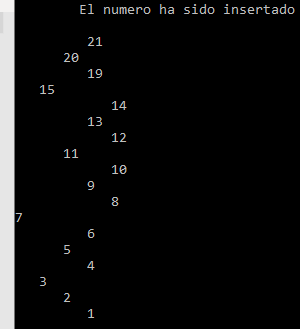


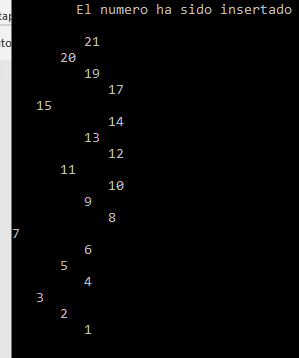


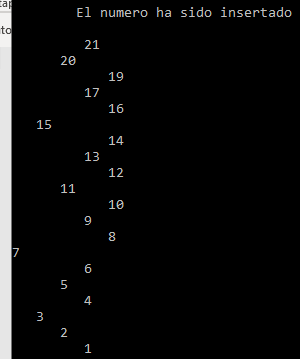


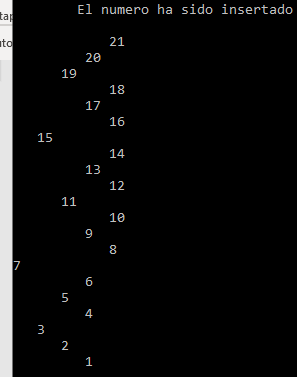












1. Supongamos que nos dan los siguientes números en ese orden 20, 10, 30, 40, 27, 35, 40, 50, 60, 67, 80, 62, 70, 83, 84, 85

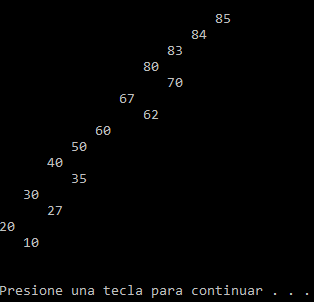
a. Si se guardan en una lista cuantos saltos se deben hacer para llegar

a 85

16 saltos en una lista

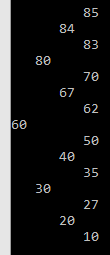
b. Si se Guardan en un ABB cuantos saltos se deben hacer para llegar a

85

 10 Saltos en ABB

c. Si se Guardan en un AVL cuantos saltos se deben hacer para llegar a

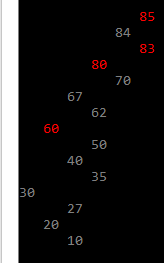
85



4 Saltos en AVL

d. Si se Guardan en un Rojo Negro cuantos saltos se deben hacer para

llegar a 85



6 Saltos en Rojo Negro

1. Supongamos que nos dan los siguientes números en ese orden 50, 55, 40, 60, 62, 51, 20, 10, 8, 9, 15, 5, 2, 3

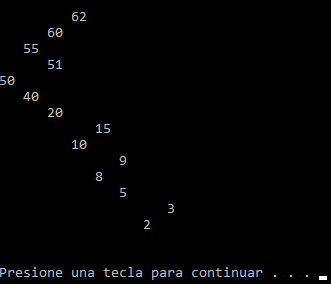
a. Si se guardan en una lista cuantos saltos se deben hacer para llegar

a 3

14 Saltos en una lista

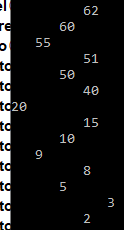
b. Si se Guardan en un ABB cuantos saltos se deben hacer para llegar a

3

 8 saltos en ABB

c. Si se Guardan en un AVL cuantos saltos se deben hacer para llegar a

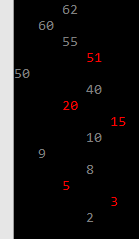
3



5 saltos en AVL

d. Si se Guardan en un Rojo Negro cuantos saltos se deben hacer para

llegar a 3



5 saltos en Rojo negro

1. Supongamos que nos dan los siguientes números en ese orden 10, 8, 15, 18, 14, 20, 3, 4, 2, 9, 20, 25, 7, 16, 29, 28, 26

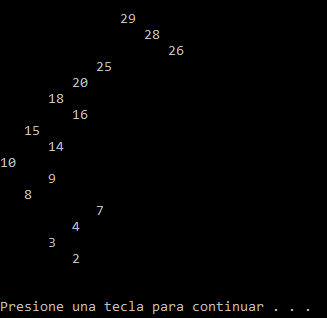
a. Si se guardan en una lista cuantos saltos se deben hacer para llegar

a 28

16 saltos en una lista

b. Si se Guardan en un ABB cuantos saltos se deben hacer para llegar a

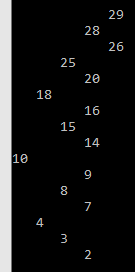
26



8 Saltos en ABB

c. Si se Guardan en un AVL cuantos saltos se deben hacer para llegar a

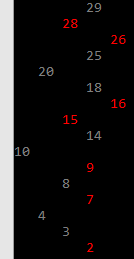
26



5 saltos en AVL

d. Si se Guardan en un Rojo Negro cuantos saltos se deben hacer para

llegar a 26



5 Saltos en rojo negro