#include <stdio.h>

#include <stdlib.h>

typedef struct No{

int info ;

struct No \*dir,\*esq;

}Tno;

void Inserir(Tno \*\*Arvore,int nun){

Tno \*novo,\*aux;

novo = (Tno \*) malloc(sizeof(Tno));

novo -> info = nun;

novo->esq = NULL;

novo->dir = NULL;

if(\*Arvore == NULL){

\*Arvore = novo;

}else{

aux = \*Arvore;

while(aux != NULL){

if(nun < aux ->info){

if(aux->esq != NULL)

aux = aux->esq;

else{

aux ->esq = novo;

return;

}

}

else{

if(aux->esq != NULL)

aux = aux->dir;

else{

aux ->esq = novo;

return;

}

}

}

}

}

void InserirSemRepitidos(Tno \*\*Arvore, int nun){

Tno \*novo,\*aux,\*aux2;

if(\*Arvore == NULL){

novo = (Tno \*) malloc(sizeof(Tno));

novo->info = nun;

novo->esq = NULL;

novo->dir = NULL;

\*Arvore = novo;

}

else{

aux = (\*Arvore);

while(aux != NULL){

if(nun == aux->info){

printf("Valor repetido!\n");

return;

}

else if(nun < aux->info){

if(aux->esq != NULL)

aux = aux->esq;

else{// -- Quando for Menor que aux->info (Esquerda) --

novo = (Tno \*) malloc(sizeof(Tno));

novo->info = nun;

novo->esq = NULL;

novo->dir = NULL;

aux -> esq = novo;

}

}

else{// -- Quando for Maior que aux->info (Direita) --

if(aux->dir != NULL)

aux = aux->dir;

else{

novo = (Tno \*) malloc(sizeof(Tno));

novo->info = nun;

novo->esq = NULL;

novo->dir = NULL;

aux -> dir = novo;

}

}

}

}

}

Tno \*Buscar(Tno \*Arvore, int nun){ // -- Retorna o Nó se encontrar | Retorna NULL se ñ encontrar --

if(Arvore == NULL){

printf("Arvore Vazia!\n");

}

else{

while(Arvore != NULL){

if(Arvore->info == nun)

return Arvore;

else if(nun < Arvore ->info)

Arvore = Arvore->esq;

else

Arvore = Arvore->dir;

}

return NULL;

}

}