

Trabajo práctico #12

Árboles binarios de búsqueda (BSTs)

Estructuras de Datos, Universidad Nacional de Quilmes

1 de diciembre de 2021

En este trabajo se busca implementar algunas funciones de árboles binarios de búsqueda (BSTs) usando el lenguaje C/C++. Para eso, se define la siguiente interfaz:

```
#include <iostream>
using namespace std;

struct BSTHeader;
typedef BSTHeader* BST;

BST emptyBST();
void InsertBST(string k, string v, BST t);
string lookup(string k, BST t);

void ShowBST(BST t);
```

Se provee además ya implementados el tipo de representación y algunas de las funciones.

```
#include <iostream>
#include <string.h>
using namespace std;

struct MapNodeSt {
    string    key;
    string    value;
    MapNodeSt* left;
    MapNodeSt* right;
};

struct BSTHeader {
    MapNodeSt* root;
};

typedef BSTHeader* BST;

BST emptyBST() {
    BSTHeader* t = new BSTHeader;
    t->root = NULL;
    return t;
}

string lookup(string k, BST t) {
    // COMPLETAR
    return "COMPLETAR";
    // COMPLETAR
}
```

```

}

void InsertBST(string k, string v, BST t) {
    // COMPLETAR
    return "COMPLETAR";
    // COMPLETAR
}

void Indent(int n) {
    for(int i=0; i<n; i++) cout << " ";
}

void ShowTreeAt(MapNodeSt* t, int i) {
    if (t==NULL) return;
    if (t->right!=NULL)
        { ShowTreeAt(t->right, i+1); }
    Indent(i); cout << " " << t->key << " := " << t->value << endl;
    if (t->left!=NULL)
        { ShowTreeAt(t->left, i+1); }
}

void ShowBST(BST t) {
    cout << "Tree(" << endl;
    ShowTreeAt(t->root, 5);
    cout << " )";
}

```

El trabajo consiste en implementar las funciones que faltan. Esa implementación tiene que poder usarse con el siguiente programa de prueba,

```

#include <iostream>
using namespace std;
#include "BST.h"

int main() {
    BST t = emptyBST();
    InsertBST("L", "1", t);
    InsertBST("G", "2", t);
    InsertBST("J", "5", t);
    InsertBST("K", "11", t);
    InsertBST("D", "4", t);
    InsertBST("S", "3", t);
    InsertBST("P", "6", t);
    InsertBST("M", "12", t);
    InsertBST("B", "8", t);
    InsertBST("E", "4", t);
    InsertBST("V", "7", t);
    InsertBST("U", "14", t);
    InsertBST("Y", "15", t);
    ShowBST(t);
}

```

La entrega debe consistir de un archivo .zip conteniendo tres archivos con nombres BST.h, BST.cpp y PruebaBST.cpp correspondiendo al código explicado previamente, y de tal forma que al compilarlos con g++ se genere un ejecutable que imprima la siguiente salida:

```
Tree(  
    Y := 15  
    V := 7  
    U := 14  
    S := 3  
    P := 6  
    M := 12  
    L := 1  
    K := 11  
    J := 5  
    G := 2  
    E := 4  
    D := 4  
    B := 8  
)
```

El archivo debe llevar por nombre 2021s2-EstrD-TP-<ApellidoNombre>.zip y ser enviado al mail de docentes. Las entregas que no respeten este formato no serán tenidas en cuenta. Se les enviará un mail con la recepción del trabajo cuando el mismo haya sido recibido correctamente.