

Laboratorio No. 4: Árboles binarios

Cristian Camilo Rendón Cardona

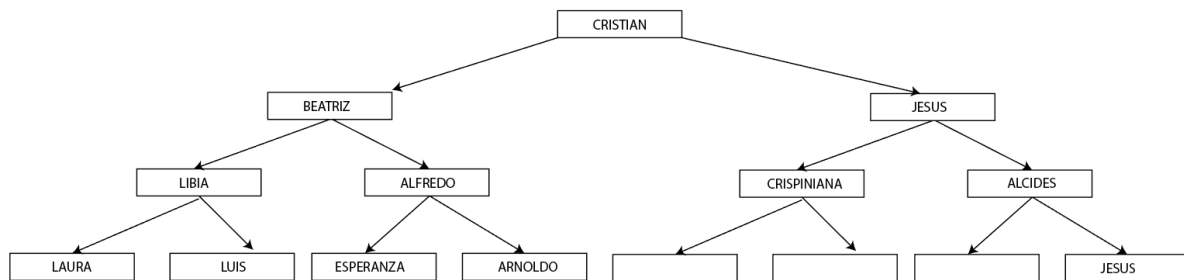
Universidad Eafit
Medellín, Colombia
crendo11@eafit.edu.co

Jhesid Steven Suarez Berrio

Universidad Eafit
Medellín, Colombia
jssuarezb@eafit.edu.co

1) Ejercicios para entregar en GitHub

1. Árbol familiar



3) Simulacro de preguntas de sustentación de Proyectos

1. No es posible implementar una búsqueda que trabaje en tiempo logarítmico porque no es posible identificar si un nombre es de mujer o de hombre, por lo tanto es necesario buscar en todo el árbol.
2. Para el ejercicio 2.1 se implementaron los algoritmos pos-orden y pre orden así:

DOCENTE MAURICIO TORO BERMÚDEZ

Teléfono: (+57) (4) 261 95 00 Ext. 9473. Oficina: 19 - 627

Correo: mtorobe@eafit.edu.co

```
private void PreOrden(Nodo node) {  
    if (node != null) {  
        System.out.println(node.dato);           // c1  
        PreOrden(node.izquierda);                // c2 + T(n/2)  
        PreOrden(node.derecha);                  // c3 + T(n/2)  
    }  
}  
  
private void PosOrden(Nodo node) {  
    if (node != null) {  
        PosOrden(node.izquierda);                // c1 + T(n/2)  
        PosOrden(node.derecha);                  // c2 + T(n/2)  
        System.out.println(node.dato);           // c3  
    }  
}
```

El algoritmo pre orden, imprime la raíz del árbol, y luego hace un llamado recursivo al sub árbol derecho para imprimir su raíz, hasta que termina y luego pasa al sub árbol derecho más profundo para imprimir su raíz, así se devuelve hasta entrar al sub árbol derecho para realizar el mismo procedimiento.

El algoritmo pos Orden Primero se va hasta el valor más profundo de la izquierda y se devuelve imprimiendo los valores más profundos de cada sub árbol y luego su raíz.

3. Ambos algoritmos realizan la misma cantidad de operaciones

$$T(n) = c + 2T(n/2)$$

Resolviendo el wólfam

$$T(n) = c(n-1) + (c1*n)/2$$

Por lo tanto la complejidad de los algoritmos es:

$$O(n)$$

4. Para el cálculo de la complejidad de los algoritmos de impresión, solo se ve la variables n representa el número de nodos del árbol binario al que se le quiere hacer la impresión.

4) Simulacro de Parcial

1. a. $1 + altura(raiz.izq);$
b. $1 + altura(raiz.der);$
2. c
3. a. false
b. $return suma == 0;$
c. $sumaElCamino(a.izq, suma - a.data);$
d. $sumaElCamino(a.der, suma - a.data);$

4.

4.1. *b*

4.2. *a*

4.3. *d*

4.4. *b*

5. *a. (toInsert >= p.left) && (toInsert <= p.rigth)*
 b. toInsert > p.left

6.

6.1. *a*

6.2. *return 0;*

6.3. *==0*

7.

7.1. *2*

7.2. *2*