ESTRUCTURA DE DATOS 1
Código ST0245

Laboratory practice No. 4: Trees

Juan Sebastian Diaz Osorio
Universidad Eafit
Medellín, Colombia
jsdiazo@eafit.edu.co

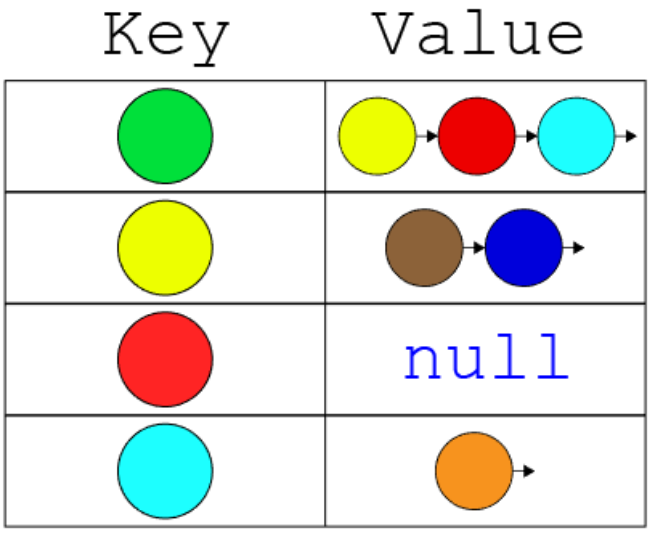
Liz Oriana Rodrigues Cruz
Universidad Eafit
Medellín, Colombia
lorodriguc@eafit.edu.co

1)

2)

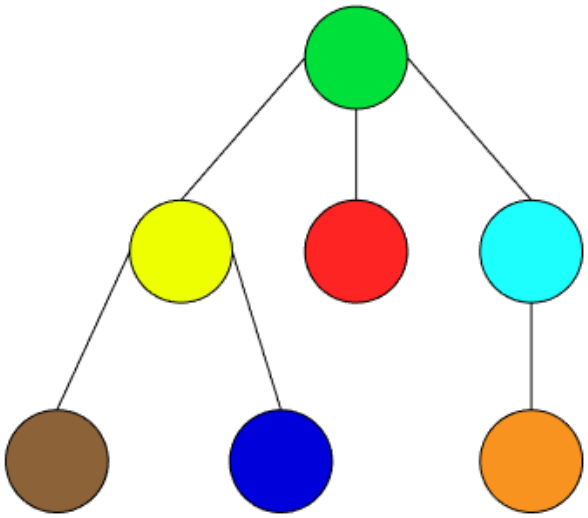
3) Practice for final project defense presentation

3.1 According to fastest search necessity, we decided to use a Hash Table with Nodes. The key is a File represented as a Node (with name, user, weight and container where it is) and the stored value is a LinkedList with its sons or null if it does not have:



PhD. Mauricio Toro Bermúdez
Professor | School of Engineering | Informatics and Systems
Email: mtorobe@eafit.edu.co | Office: Building 19 – 627
Phone: (+57) (4) 261 95 00 Ext. 9473

Deeply, this structure represents a n-ary tree, but search operation is **O(n)**. Previous Hash Table would be like this:



To solve our problem, we create a progressive searching. When user puts “main/src/img”, first, search algorithm checks if main is the root bin, then, looks for any son of “main” that is called “src” and take this one as root. Now, by recursion, it asks for any “src” son that is called “img”... It continues until it reaches last bin, “img” in our example. Here, all “img” sons’ names are printed (if this one has sons).

3.2 This question was answered in previous point. Searching in a tree is so efficient if we implement it as Hash Table.

3.3 The implementation of the code that receives a pre-order binary tree to a tour of a post-order binary tree is very simple, although you need to be aware of the details, we must use two classes, the main class (Tree) that is responsible for Perform the entire process of change and structure, while the second class is already secondary and not as relevant as it is only responsible for receiving the pre-order binary tree and executes the necessary methods for the change.

PhD. Mauricio Toro Bermúdez
Professor | School of Engineering | Informatics and Systems
Email: mtorobe@eafit.edu.co | Office: Building 19 – 627
Phone: (+57) (4) 261 95 00 Ext. 9473

The Tree class consists of 7 methods in total, each of which handles a special part for the change, although the most important for the process to be carried out successfully are the *PreOrderRecursion* methods which is responsible for reading and walking the tree with the help of Stack in the event that something has been entered incorrectly, reorganizes it by asking if there are still elements to identify by means of a "*while*" cycle and asking "*if / else*" questions. The heart of the code for it to work successfully is the *PostOrdenRecursion* method which performs a procedure very similar to the *PreOrdenRecursion* using "*Stack*" batteries, "*while*" cycles and "*if / else*" questions but with the difference that as this method if you need to make a change yes or if you need to be much more detailed when asking.

3.4 and **3.5**

Method	Complexity	
PreOrdenRecursion	O(log n)	is the length of the chain in the stack
PostOrderRecursion	O(log n)	is the length of the chain in the stack

4) Practice for midterms

4.1 Answers:

4.1.1 line 04: `1 + altura(raiz.izq);`

4.1.2 line 05: `1 + altura(raiz.der);`

4.2 c

4.3 Answers:

4.3.1 line 03: `return false;`

4.3.2 line 05: `return suma == a.dato;`

4.3.3 line 07: `sumaElCamino(a.izq, suma - a.dato);`

PhD. Mauricio Toro Bermúdez
Professor | School of Engineering | Informatics and Systems
Email: mtorobe@eafit.edu.co | Office: Building 19 – 627
Phone: (+57) (4) 261 95 00 Ext. 9473

4.3.4 line 08: `sumaElCamino(a.der, suma - a.dato);`

4.4 Answers:

4.4.1 c

4.4.2 c

4.4.3 d

4.4.4 a

4.5 Answers:

4.5.1 line 04: `if (p.value == toInsert)`

4.5.2 line 06: `if (p.value > toInsert)`

4.6 Answers:

4.6.1 c

4.6.2 line 04: `return 0;`

4.6.3 line 06: `if (raiz.hijos.size() == 0);`

4.7 Answers:

4.7.1 a

4.7.2 b

4.8 Confusing question. We can not understand it

4.8 Sean *A* y *B* las salidas de los recorridos pre-orden y pos-orden del árbol binario anterior, respectivamente. Determinen el número de elementos para los cuales se cumple que para $1 \leq i \leq 8$.

Elijan la respuesta que consideren acertada:

a) *3*
b) *2*
c) *4*
d) *0*

.9 a

10 b, no. Some nodes have left and right sons bigger than themselves.

4.11 Answers:

4.11.1 b

4.11.2 a

4.11.3 a

PhD. Mauricio Toro Bermúdez
Professor | School of Engineering | Informatics and Systems
Email: mtorobe@eafit.edu.co | Office: Building 19 – 627
Phone: (+57) (4) 261 95 00 Ext. 9473

4.12 Answers:

4.12.1 i

4.12.2 Wrong terms. “Pos-order” would be “A, C, B, F, E, D, H, J, I, G” Maybe teacher would want to say “in-order” according to possible answers. In that case, answer is b

4.12.2 ¿Cuál sería el recorrido pre-orden del árbol anterior, asumiendo que siempre se visita primero la hoja de la izquierda?

a) *G, D, B, A, C, E, F, I, H, J*
b) *A, B, C, D, E, F, G, H, I, J*
c) *B, C, D, E, F, G, H, I, J, A*
d) *A, C, B, D, I, J, H, E, F, G*

4.13 Answers:

4.13.1 line 10: `suma[raíz.id] = suma[e.id] + suma[raíz.id];`

4.13.2 d

5) Recommended reading (optional)

https://prezi.com/fjdflicgg_6z/?utm_campaign=share&utm_medium=copy

6) Teamwork and gradual progress (optional)

PhD. Mauricio Toro Bermúdez
Professor | School of Engineering | Informatics and Systems
Email: mtorobe@eafit.edu.co | Office: Building 19 – 627
Phone: (+57) (4) 261 95 00 Ext. 9473