

Laboratory practice No. 4: Binary trees and hash tables

Gregorio Pérez Bernal
Universidad Eafit
Medellín, Colombia
Gperezb1@eafit.edu.co

Luisa Toro Villegas
Universidad Eafit
Medellín, Colombia
ltorov@eafit.edu.co

3) Practice for final project defense presentation

3.1 in order to implement the exercise it was found that a binary tree wouldn't solve the problem because it was not certain that every node would have two "sons", therefore, an n-ary tree was implemented, and it is expected to have a complexity of $O(n^2)$ because there is a for loop ($O(n)$) with a recursive call ($O(n)$) nested.

3.2 It is not possible because there is no way to organize it that will aid to shorten the time of search.

3.4

In order to analyze the complexity, one must take into account the complexity of insertion in a binary tree which is $O(\log n)$. However, in the method "posOrden" There are two recursive calls which by doing the complexity analysis, we get a complexity of $O(2^n)$, meaning that the overall complexity for the worst case, due to the addition rule is of 2^n .

3.5 n refers to the number of nodes in the tree

4) Practice for midterms

4.1 B
D

4.2 False
a.data
a.izq, suma-a.data
a.der, suma-a.data

4.6 D
Return 0;

ESTRUCTURA DE DATOS 1
Código ST0245

== 0
4.7 A
B
4.8 B
4.9 A
4.10 B
4.11 B
A
B
4.12 A
A
B

4.13 raiz.id
A

5) Recommended reading (optional)

ANNEXED AS A PDF

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