# **Laboratory practice No. 4: Tablas de Hash y Arboles**

|  |  |
| --- | --- |
| **Julian David Ramirez Lopera**  Universidad Eafit  Medellín, Colombia  jdramirezl@eafit.edu.co | **Andres Salazar Galeano**  Universidad Eafit  Medellín, Colombia  asalaza5@eafit.edu.co |

**3) Practice for final project defense presentation**

**3.1**

**3.4 Time complexity for this algorithm is O(n)**

**3.5 n stands for the the number of nodes in the binary tree. In this case each node is visited twitce. However, the overall time complexity remains at O(n)**

***4) Practice for midterms***

* 1. *We used Octrees*, they have a insert and search time complexity of O(logn)
  2. *Returns the node with children equal to n1 and n2 or in case that condition is not met, it returns the node that has one child equal to n1 or n2. / O(n) where n is the number of nodes in the binary tree / the values of the children can be compared to n1 and n2 respectively, and dicard paths which are not in the range of possible nodes having n1 and n2 as children.*
  3. *return True/ O(n+m)*
  4. *A / A / D / A / (4) p.val == toInsert | (6) toInsert < p.val*