Collisions between robotic bees: a data structure on how detect them efficiently.

Sebastián Arboleda.
Carla Rendón.
Medellín, 15 of may 2018

Design Data Structures

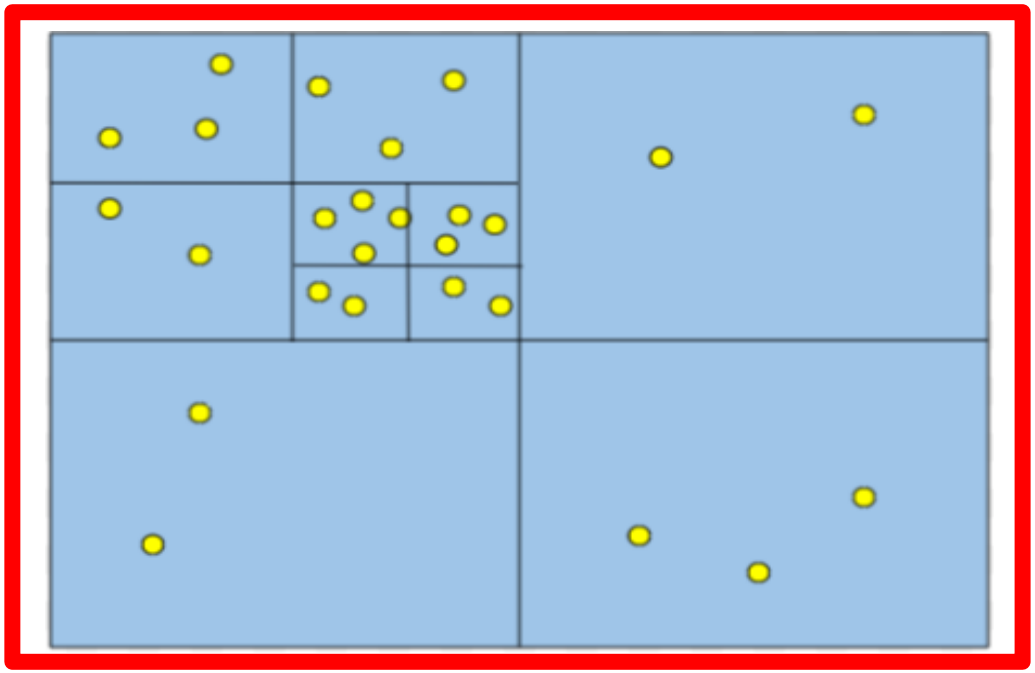


Figure 1: Quadtree with bees inside
Each node is divided into four parts when more than 10 bees are inserted in the same square (node)

Data Structure Operations

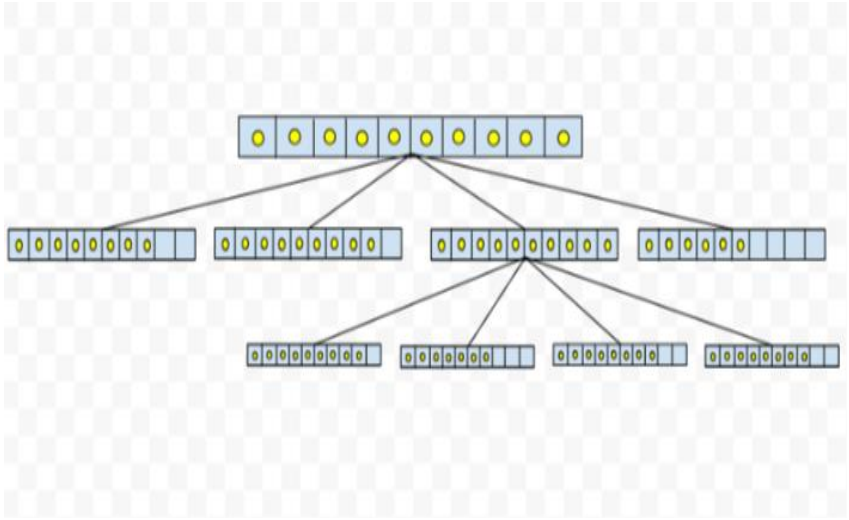


Figure 2: Division of tree nodes through arrays

Operations	Complexity.
Lector	O(n)
Transform Strings into files	O(n/2)
Quadtree	O(Log4(n)
Insertion of all bees	O(n)
Comparison of bees	O(10^2)

Table 1: Complexity of operations of the data structure

Design Criteria of the Data Structure

For the realization of this algorithm we consider the complexity, since when carrying out a program with inputs like reality (10,000,000 of bees, for example), it is necessary that the operations are carried out quickly and effectively, the algorithm known as Quadtree offers a complexity O (log4 (n)), which allows great speed implementing operations within the program. On the other hand, we decided to also implement an ArrayList within each node of the tree, to clearly define when a new division in the plane must be made, in our case each ArrayList has a maximum of 10 spaces, when an eleventh bee tries to enter the node, this is divided into 4 new ArrayList (nodes).

Design Criteria of the Data Structure

Firts solution:

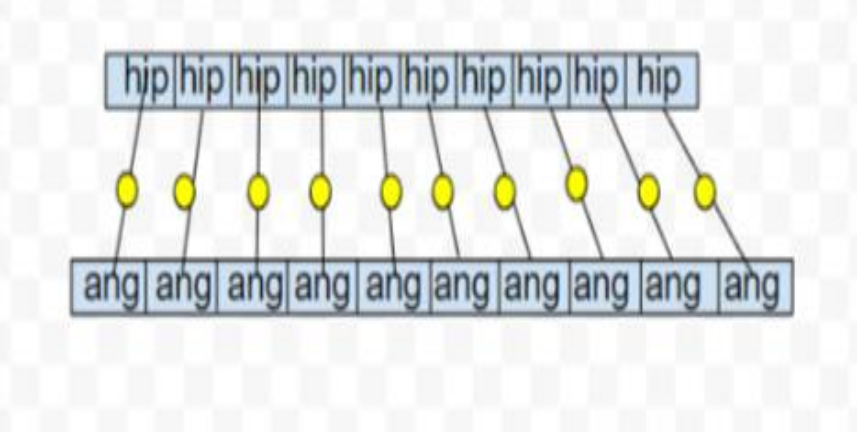


Figure 3: Arrays and Bees.

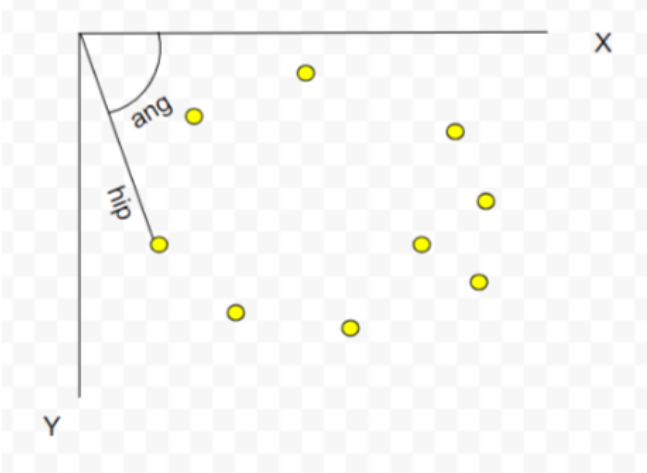
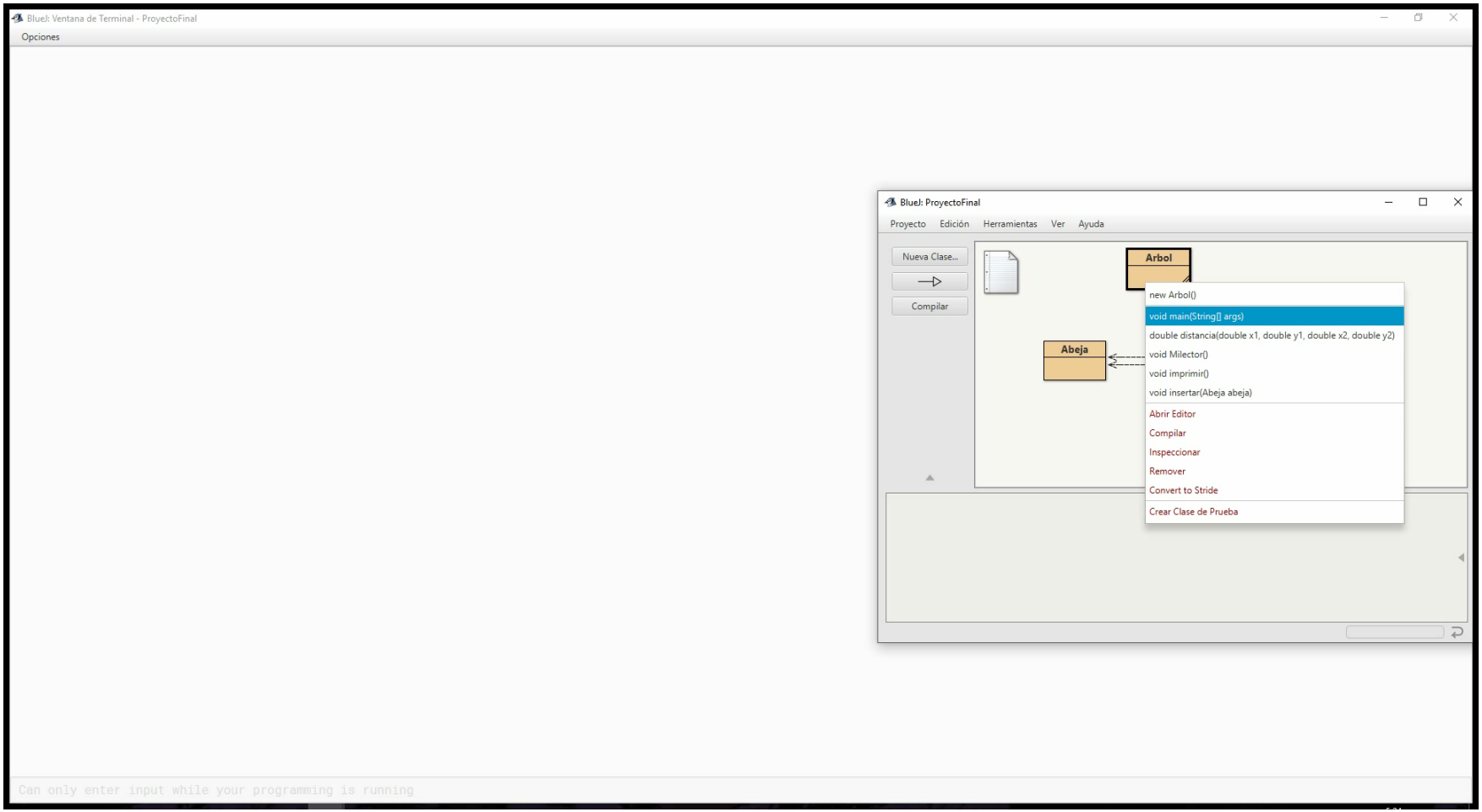


Figure 4: Hipothenuse and angle formed by the bees.

Time and Memory Consumption

10	6,462048
10	6,462048
100	9,919091
1000	19,82767
10000	165,345268
100000	955,2793479
1000000	6.493

Implementation



THANKS FOR YOUR ATTENTION