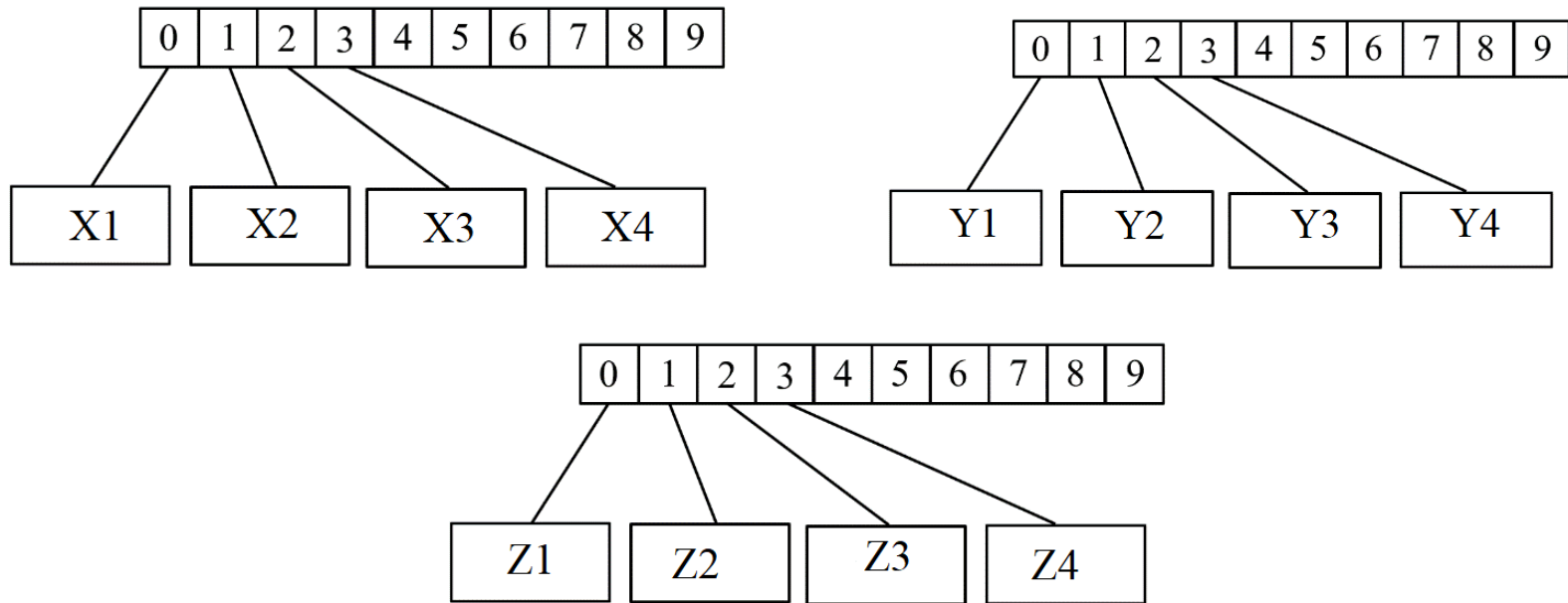


TECHNOLOGY AND ALGORITHMS: A SOLUTION TO BEE´S EXTINTION

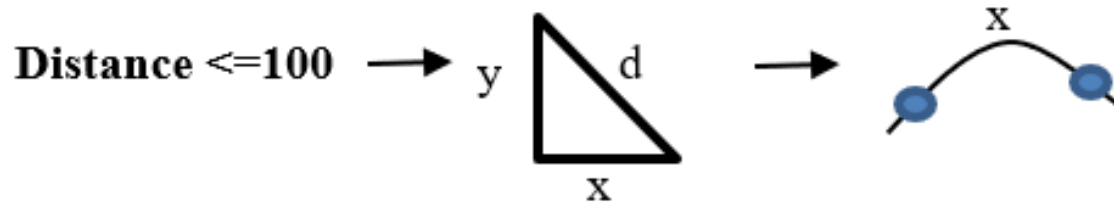
*Isabella Arango Restrepo
Juan David Rengifo Castro
Medellín, November 06*

First Data Structure Designed



Graphic 1: ArrayList of bee's coordinates. A coordinate is double number that represent the latitude, longitude or height.

Data Structure Operations



Graphic 2: Close bees operation with array lists. It compare two coordinates and verify if the distance between them is less than 100m.

Method	Complexity
CloseBees	$O(n^2)$

Table 1: Complexity of the operations of ArrayLists.

Design Criteria of the Data Structure

- To solve the problem it was necessary to test it with groups of data of different sizes, that is to say that this was not a constant factor.
- The data structure used should be one to which you could add many data and obtain each of its specifications (length, latitude, height).
- The add and get operation in an ArrayList have time complexity of $O(1)$, so they were very efficient no matter the size of the data.

Time and Memory Consumption

Operations	10 bees	100 bees	1000 bees	10000bees
closeBees	3.33ms	23.5ms	394.67ms	23073.67ms

Table 2: Execution time for the method closeBees for different data sets.

	10 bees	100 bees	1000 bees	10000 bees
Memory used	2MB	3MB	4MB	7MB

Table 3: Memory used by different data sets.

Implementation

```
Bee #: 9756 and bee #: 9825 are at 95.36481225782188m.
Bee #: 9760 and bee #: 9778 are at 9.984817737735854m.
Bee #: 9776 and bee #: 9869 are at 82.95451199572m.
Bee #: 9780 and bee #: 9971 are at 98.03196478000936m.
Bee #: 9783 and bee #: 9938 are at 82.05503729468987m.
Bee #: 9789 and bee #: 9893 are at 97.83764463457703m.
Bee #: 9790 and bee #: 9890 are at 88.9715001086929m.
Bee #: 9794 and bee #: 9862 are at 97.65983312687356m.
Bee #: 9796 and bee #: 9808 are at 31.145866102527247m.
Bee #: 9796 and bee #: 9913 are at 97.39101155051945m.
Bee #: 9808 and bee #: 9913 are at 67.54633597178817m.
Bee #: 9814 and bee #: 9826 are at 88.14724400730923m.
Bee #: 9815 and bee #: 9843 are at 56.47192610243705m.
Bee #: 9821 and bee #: 9884 are at 47.089576835808195m.
Bee #: 9828 and bee #: 9977 are at 91.89673160472923m.
Bee #: 9833 and bee #: 9873 are at 99.87646649811163m.
Bee #: 9839 and bee #: 9990 are at 51.941878654384006m.
Bee #: 9842 and bee #: 9933 are at 85.2757801650051m.
Bee #: 9842 and bee #: 9940 are at 33.42387965690543m.
Bee #: 9843 and bee #: 9891 are at 92.3996039545545m.
Bee #: 9844 and bee #: 9944 are at 97.28517296973513m.
Bee #: 9867 and bee #: 9931 are at 93.19547494277548m.
Bee #: 9900 and bee #: 9982 are at 83.70826833887558m.
Bee #: 9901 and bee #: 9953 are at 84.85695457531075m.
Bee #: 9933 and bee #: 9940 are at 96.91435219019583m.
Bee #: 9977 and bee #: 9982 are at 70.55791587971176m.
```

```
Memoria máxima: 247MB
Memoria total: 15MB
Memoria libre: 10MB
Memoria usada: 5MB
El tiempo de ejecución es: 20527
```

```
Project.main({ });
```

```
Bee #: 3 and bee #: 9 are at 72.97838065101487m.
```

```
Bee #: 6 and bee #: 8 are at 73.17524589460847m.
```

```
Memoria máxima: 247MB
```

```
Memoria total: 15MB
```

```
Memoria libre: 9MB
```

```
Memoria usada: 6MB
```

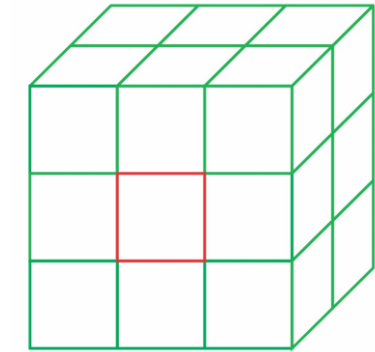
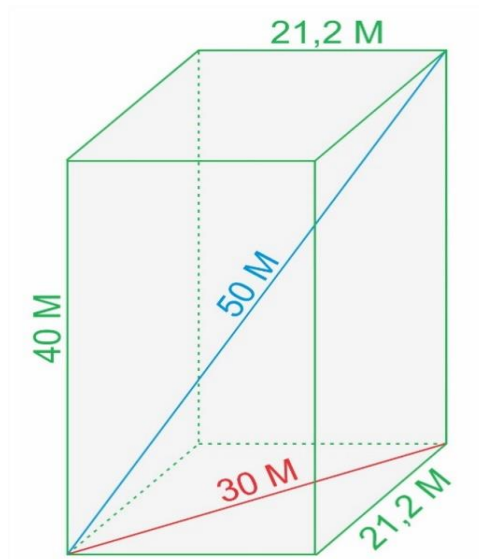
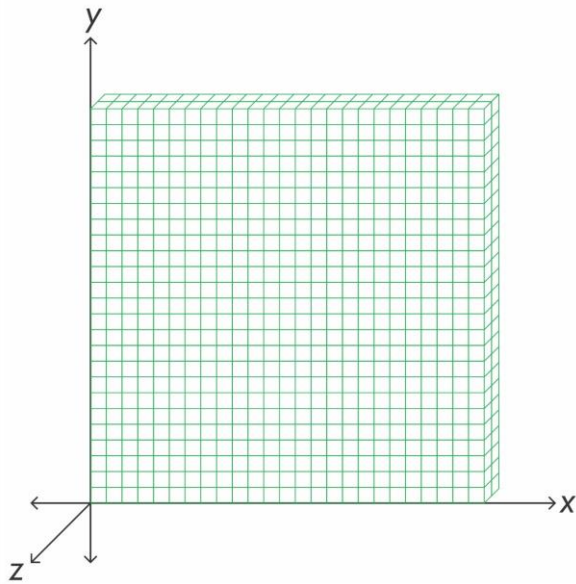
```
El tiempo de ejecución es: 2
```

Graphic 3: Bees that have a distance of less than 100m between them.

Second Data Structure Designed

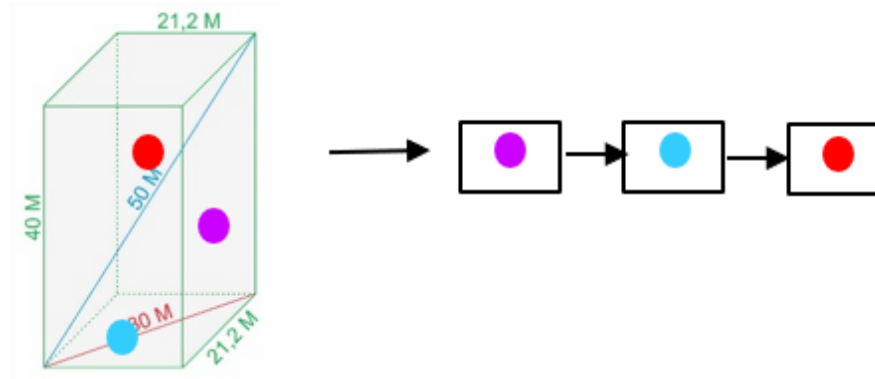


Graphic 4: Map of Bello with its maximum and minimum coordinates



Graphic 5: Representation of the map divided by parallelepipeds of respective measures.

Hash Table Operation



Graphic 6: Addition of bees that are in the same parallelepiped to a LinkedList.

Design Criteria of the Data Structure

- To reduce the time complexity of the first algorithm it was needed to choose another data structure that works well with larger data samples.
- Any operation of the data structure (Hash Table) can be done in constant time, which allows having an efficient algorithm.
- To continue with such low complexity, it was necessary to choose a good data structure to organize the elements in the same parallelepiped, taking into account that the number is not constant.

References

LA VANGUARDIA. Available from <https://www.lavanguardia.com/natural/20161005/41771284333/abeja-peligro-humanos.html> (2016); accessed 24 August 2018.

Runestone. Available from <http://interactivepython.org/runestone/static/pythoned/SortSearch/TransformacionDeClaves.html> (2012); accessed 24 August 2018.

Azure from the trenches. Available from: <https://www.azurefromthetrenches.com/introductory-guide-to-aabb-tree-collision-detection/> (2017); accessed 24 August 2018.

Wikipedia. Available from <https://en.wikipedia.org/wiki/R-tree> (2018); accessed 24 August 2018.

Enrique Munguía. Available from <http://www.enrique7mc.com/> (2018); accessed 22 August 2018

- Evantotuts+. Available from <https://gamedevelopment.tutsplus.com/tutorials/quick-tip- use-quadtrees-to-detect-likely-collisions-in-2d-space-- gamedev-374> (2018); accessed 24 August 2018.
- Wikipedia. Available from https://en.wikipedia.org/wiki/Quadtree#Some_common_uses_of_quadtrees (2018); accessed 24 August 2018.
- The IUCN Red List of Threatened Species. Available from <http://www.iucnredlist.org/news/nearly-one-in-ten-wild-bee-species-face-extinction-in-europe-while-the-status-of-more-than-half-remains-unknown-iucn-report> (2018); accessed 24 August 2018.
- Big-O Cheat Sheet. Available from <http://bigocheatsheet.com/> (2018); accessed 20 August 2018.
- Information Technology Gems. Available from <http://infotechgems.blogspot.com/2011/11/java-collections-performance-time.html> (2018); accessed 20 August 2018.