COMP3010 Ass1 Report

For the first part of my algorithm where the objective is to interpret and process the users’ inputs of representatives and put them into a HashMap, this involves scanning the System.in and identifying each Integer until we reach the end of the desired array, which is indicated by the integer “0”.

Since we are looking at every input once and only once, and no sorting of any kind needs to be done, we know that this meant the Big Oh notation for the efficacy of this process is O(n) where n is the size of the desired input.

For the 2nd part of my assignment, we are looking at the actual algorithm which will take the HashMaps as inputs and then determine the smallest (non 0) number of members which can be included as the final output. I have achieved this by placing the list of members into the arrayLists and HashMaps which will then be run through incrementally until every input has been observed and if the next number if smaller than the previous one, then it is set to be the new current, otherwise the old current number is kept.

Since we are sorting using HashMaps which stores key-value pairs and we are comparing the values with each other and is a self-balancing binary tree, and when you sort them, you end up with a worst case time complexity of O(n log n). The space complexity is O(n) since we are storing n items in our HashMaps.