Method 1:

Data structure used : Hash table (unorder\_map)

Algorithm

-Traverse the list of input numbers, count them into the map : O(n)

-Traverse the key of map, find out the key with max value(count) : O(n)

Worst case time complexity for each test case : O(n)

一張含有 文字, 螢幕擷取畫面, 軟體, 多媒體軟體 的圖片

自動產生的描述

Method 2:

Data structure used : LinkedList

Algorithm

-Group the input numbers together by is value through insert them into LinkedList : O(n)

-traverse the LinkedList, find out the most frequent duplicate : O(n)

Worst case time complexity for each test case : O(n)

一張含有 文字, 螢幕擷取畫面, 軟體, 多媒體軟體 的圖片

自動產生的描述

Method 3:

Data structure used : binary search tree

Algorithm

- Traverse the list of input numbers, count them into the binary search tree, number with the same value of root add to left : O(n)

- Traverse the tree by inorder, find out most frequent duplicate, when there are multiple most frequent duplicate, traverse the input numbers to find out the one first exists in the input numbers : O(n^2)

Worst case time complexity for each test case : O(n^2)

一張含有 文字, 螢幕擷取畫面, 軟體, 多媒體軟體 的圖片

自動產生的描述