Practice M4

Phuc Tran (Jennifer)
MSWE - Fall 2021
241P - Applied Data Structures & Algorithms
The Algorithms Design Manual - Second Edition
Homework · 4-5 , 4-16 , 4-40

4.5 The mode of a set of numbers is the number that occurs most frequently in the set. The set (4, 6, 2, 4, 3, 1) has a mode of 4. Give an efficient & correct algorithm to compute the mode of a set of n numbers.

Sort the set first.

Then scan the set and keep a counter for the number w/ the max frequency

→ O(nlogn) time O(nlogn) + O(n) = O(nlogn) (pick the bigger one)

4-16 Use the partitioning idea of quick sort to give an algorithm that finds the median element of an array of n integers in expected O(n) time.

Hint: must you look at both sides of the partition?

- Choose a pivot element
- Split the array into 2 parts (subarrays)
- Use the partition step from the Quicksort algorithm to arrange all elements that are smaller than the pivot to the left of it, and, put all the elements that are greater than the pivot to the right of the pivot. → this takes O(n) time.
- If the pivot is in the middle, pivot is the median
- If not, repeat the step with the small values array, with the pivot as the ending element of that subarray.

If the position of the pivot is after the middle element, then repeat the procedure for the subarray w/ the pivot as the first element. → this takes O(n/2) time

Repeat takes O(n/4), O(n/8), ... → total is O(n) time

4-40 If you are given a million integers to sort, what algorithm would you use to sort them? How much time & memory would that consume?

The range of the array is known ~ 1 million

Input array A with n elements

Counting array B with k elements

→ use counting sort

Running time O(n + k)