

Data Structures and Algorithms Homework 2

Due Wednesday Sept 11; Joseph Sepich (jps6444)

1 Problem 1 Sorted Array

Given a sorted array A of n (possibly negative) distinct integers, you want to find out whether there is an index i for which $A[i] = i$. Give a divide-and-conquer algorithm that runs in time $O(\log n)$. Provide only the main idea and the runtime analysis.

There are two very important parts to this problem. The first important part of the problem is the fact that the array is sorted. The second important part is knowing that you are looking for a integer contained in the array that is the same value as the index. From here we can use a binary search like algorithm. If we look at the middle integer and the statement holds, $A[i] == i$, then we can return true. If the middle index is larger than the value contained in it, you know that none of the values below that index will work, because the index has to be larger than it. If the middle index is smaller than the value contained in the array at the position, then you know that none of the value above that index will work, because the index would have to be smaller than it. These two conditions also rely on the fact that A only contains integer values and not doubles or floating point values.

Knowing these three conditions you can recursively call one subproblem of size $n/2$ with constant time at each step eventually reaching a number that fulfills the requirement or a problem with size 1, that doesn't fulfill the requirement where you return false. (Accessing a value at an index is constant time). Using the master theorem we know this running time would be $O(\log(n))$, since a is 1, b is 2, and d is 0.

Collaborators: None

2 Problem 2 Squaring vs Multiplying: Matrices

The square of a matrix A is its product with itself, AA .

2.1 Part a

Show that five multiplications are sufficient to compute the square of a 2×2 matrix.