

Problem Solving Workshop #13

Tech Interviews and Competitive Programming Meetup

September 17, 2016

<https://www.meetup.com/tech-interviews-and-competitive-programming/>

Instructor: Eugene Yarovoi (can be [contacted](#) through the group Meetup page above under Organizers)

More practice questions: leetcode.com, glassdoor.com, geeksforgeeks.org

Books: Elements of Programming Interviews, Cracking the Coding Interview

Have questions you want answered? Contact the instructor, or ask on [Quora](#). You can post questions and [follow the instructor](#) and other people who write about algorithms.

Try to find optimized solutions, and provide a time and space complexity analysis with every solution.

Coding / Algorithm Problem #1

You're given an array of N integers.

- (i) **[Easy]** Determine whether there exist two values in the array that have a difference of at most K .
 - (ii) **[Easy]** Determine whether there exist two values in the array that are identical, and their index differs by at most D (they are distance D apart or closer in the array).
 - (iii) Determine whether there exist two values in the array that satisfy both conditions at once: they have a difference of at most K **and** are at most D distance apart. **[Easy]** Naive solution; **[Medium]** Optimized approach that is at least close to linear time; **[Hard]** Ideal $O(N)$ solution independent of K, D .
-

Algorithm Problem #2

[Medium to Medium-Hard], depending on how much you optimize time and space

There's a storage facility with M containers. The containers are arranged in a single line from the 0-th container to the $M-1$ th, with adjacent containers being one unit of distance apart. There are two robots that can pick up an item from one container and transport it to another container. The robots take commands in the form of two integers i, j that say "transport an item from container i to container j ". When a robot receives such a command, it moves from its current position to container i , picks up an item there, moves to container j , and then stays there until it receives another command.

You're given M and a sequence of commands that must be executed in order (you know the entire sequence up-front, but cannot re-order commands, and each command must be completed by one of the two robots before the next command can start), determine which robot should execute each command so that we minimize the sum of the total distances traveled by the robots. You get to choose the starting position of both robots.

Problem Solving Tip: first try finding the min distance without caring about which robot did what.

Example Input: [(2, 5), (12, 7), (6, 14)]

Output: Robot 1 executes commands: 0, 2 [Robot 2 executes the rest]

Explanation: To minimize total distance traveled, robot 1 will start at position 2 and execute command 0. This will cost $5 - 2 = 3$. Then robot 2 will start at position 12 and execute command 1. This will cost $12 - 7 = 5$. Then robot 1 will move to position 6 (it previously was at position 5), at a cost of $6 - 5 = 1$. It will then execute command 2 at a cost of $14 - 6 = 8$. The total distance traveled by all robots will be $3 + 5 + 1 + 8 = 17$. This is the lowest that could have been obtained.

THEORY PROBLEM

[Difficulty depends on how in-depth your answer is] Suppose you need to choose an in-memory data structure for a task. The data structure will need to support map operations: get (find), insert, and delete. Let G be the number of gets you need to do over the course of your algorithm, I the number of inserts, and D the number of deletes. Which data structure would you choose if:

- (i) You have very limited space, and $G \gg I, D$
- (ii) You have very limited space, and $I \gg G, D$
- (iii) You have very limited space, and I, G, D are about the same.
- (iv) You have lots of space, and I, G, D are about the same.

The \gg symbol means “much much greater than”. Note that just how much greater one quantity is relative to another is left deliberately ambiguous -- you might actually discuss several possible solutions for each point depending on exactly where the tradeoff lies. The same applies for the phrase “very limited” -- it may mean “almost no extra space at all” or may mean “don’t use many times the space that is required by your data, like a hash table would”.

Generally speaking, in design questions, there’s always lots of ambiguity, so you would start by asking clarifying questions to the interviewer. Since you don’t have an interviewer here, assume what you think are reasonable answers to the questions you would ask.