

CSC-421 Applied Algorithms and Structures

Fall 2021-22

Instructor: Iyad Kanj

Office: CDM 832

Phone: (312) 362-5558

Email: ikanj@cs.depaul.edu

Office Hours (Office/Zoom): Mon. 4:00-5:30 & Wed. 10:00-11:30

Course Website: <https://d2l.depaul.edu/>

Assignment 1

(Due September 27)

Remarks

- For this assignment, if needed, you may assume that sorting n numbers can be done in time $O(n \lg n)$ (e.g., using Merge/Heap Sort). If you need to sort, you can directly apply such a sorting algorithm (without writing the pseudocode), and claim that it runs in $O(n \lg n)$ time, where n is the number of elements/numbers being sorted.
- When asked to give an algorithm that meets a certain time bound, you need to give the algorithm (pseudocode/description) and analyze its running time to show that it meets the required bound; giving only the algorithm is not enough to receive full credit.
- Please submit your solutions to problems 1-5 as a single PDF file. If your solutions to problems 1-5 consist of multiple files, convert all your files into a single PDF file and upload it. Please create a single “.zip” file containing all your solutions/files (code files and non-code file) for problems 1-6 and upload it on D2L.
- Please double check that you submitted the correct and complete file after uploading your (final) submission. No late resubmissions will be allowed.

1. (10 points) Given a collection of n nuts and a collection of n bolts, arranged in an increasing order of size, give an $O(n)$ time algorithm to check if there is a nut and a bolt that have the same size. The sizes of the nuts and bolts are stored in the sorted arrays $NUTS[1..n]$ and $BOLTS[1..n]$, respectively. Your algorithm can stop as soon as it finds a single match (i.e, you do not need to report all matches).
2. (15 points) Let $A[1..n]$ be an array of distinct positive integers, and let t be a positive integer.
 - (a) (5 points) Assuming that A is sorted, show that in $O(n)$ time it can be decided if A contains two distinct elements x and y such that $x + y = t$.
 - (b) (10 points) Use part (a) to show that the following problem, referred to as the 3-SUM problem, can be solved in $O(n^2)$ time:

3-SUM

Given an array $A[1..n]$ of distinct positive integers that is not (necessarily) sorted, and a positive integer t , determine whether or not there are three distinct elements x, y, z in A such that $x + y + z = t$.

3. (10 points) Let $A[1..n]$ be an array of positive integers (A is not sorted). Pinocchio claims that there exists an $O(n)$ -time algorithm that decides if there are two integers in A whose sum is 1000. Is Pinocchio right, or will his nose grow? If you say Pinocchio is right, explain how it can be done in $O(n)$ time; otherwise, argue why it is impossible.
4. (10 points) Let $A[1..n]$ be an array of points in the plane, where $A[i]$ contains the coordinates (x_i, y_i) of a point p_i , for $i = 1, \dots, n$. Give an $O(n \lg n)$ time algorithm that determines whether any two points in A are identical (that is, have the same x and y coordinates).
5. (15 points) Show how to determine in $O(n^2 \lg n)$ time whether any three points in a set of n points are collinear.
6. (40 points) Write a program that implements Graham's algorithm for computing the convex hull of a set of points S in the plane. Your algorithm should take as input the coordinates of the points in S (following the format in the input files on D2L), and should output the coordinates of the points on the convex hull of S . You can assume

that the points in S are distinct.

You have to use (a version) of one of the following three *standard* programming languages: *C*, Java, or Python. Please test your code on the test files that are uploaded on D2L (under Submissions, in the same folder as the assignment), and compare them to the posted solutions in the solution file. The grader will test your programs on the uploaded test files (text files). So make sure that your program runs correctly on the uploaded test files.

Please create a single “.zip” file containing your source code for problem 6, plus your solutions to problems 1-5 (as a single PDF file), and upload it on D2L.

7. **Suggested Programming Problems on LeetCode; not to be submitted.** Below are the titles of coding problems (related to the materials we covered so far) that I suggest that you do in LeetCode.com (under the tab “Problems”). You can test your submissions and view the solutions in LeetCode.

- (i) Two Sum.
- (ii) 3Sum.
- (iii) Valid Triangle Number.