



Algorithm Design and Analysis

Assignment 03: Decrease-and-conquer algorithms and Recurrence relations

Total Points:	40
Released:	Thursday, 14 th February 2019
Due:	Thursday, 21 st February 2019
Submission format:	Soft copy on CAMU / Hard copy to the FI

Instructions

- Submit individual solutions to this assignment. Refer to the policies regarding collaboration in the syllabus.
- For the written part of the assignment (Part A), you can either submit a hard copy to the FI (before 5pm on the due date), or upload a soft copy to CAMU. A hard copy is preferred. For Part B, upload a soft copy of your code to CAMU.
- For submissions on CAMU, first combine all documents to be submitted into a folder named with your ID number appended with "-Assign03" (e.g. 99992020-Assign03) and then zip the folder. Make sure you rename the folder *before* zipping, so that when we unzip the folder it has the correct name.

Preparation

Study Sections 2.4, 4.1, 4.2, 4.4 and 4.5 with your study partner/group. Discuss key ideas as well as things you do not understand, and bring questions to office hours.

Rather than studying the entire chapter in one sitting, you may wish to study the chapter in sections, working on the self-study questions above and assignment questions below corresponding to each section. For example, after you study section 2.4, do assignment problem 1. Then, you can move on to study section 4.1, and so on.

PART A

Problem 1 [6 points]: *Solving recurrence relations*

Exercise 2.4, #1 b, c, d. (Use the method of backward substitutions)

Problem 2 [4 points]: *Analyzing recursive algorithms*

Exercises 2.4, #3.

Problem 3 [2 points]: *Understanding insertion sort*

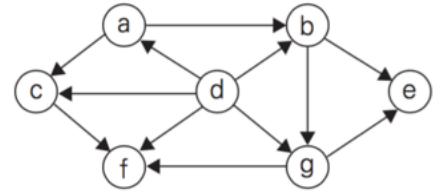
Apply insertion sort to sort the list U N I V E R S I T Y

Problem 4 [3 points]: *Ferrying soldiers*

Exercise 4.1 #1

Problem 5 [6 points]: Understanding topological sorting

- (i) Solve the topological sorting problem for the diagraph below using
- the DFS-based algorithm
 - the source removal algorithm



- (ii) Exercise 4.2, #4

Problem 6 [5 points]: *Understanding the fake-coin problem*

Exercise 4.4, #10

Problem 7 [2 points]: *Understanding the Russian peasant multiplication algorithm*

Exercise 4.4 #11

Problem 8 [2 points]: Understanding the quickselect algorithm

Apply quickselect to find the median of the list of numbers 8, 15, 3, 19, 24, 32, 6

PART B

Problem 9 [10 points]: *Implementing the quickselect algorithm*

Implement the recursive code for quickselect in the language of your choice. Do not forget to include a test program that properly demonstrates the functionality of your code

EXTRA CREDIT PROBLEMS

Extra Credit Problem 1: *A decrease-by-one algorithm for finding factors*

Exercise 4.1 #4

Extra Credit Problem 2: *Cutting a stick*

Exercise 4.4 #1