Name: Section #   CM#

CSSE/MA 473 Assignment 0
Due date: Tue Dec 1

Please read the following homework guidelines before starting.

- Get started early on all assignments! Sometimes you will need "incubation time" to work out problems.

- Assignment submissions are to be typed or written neatly.

- Use plain or lined paper. Don't use graph paper (grid lines are too dark), and if you use engineering paper, only use the front (lighter-lined) side.

- Problems must be numbered as they are on this paper (1, 2, 3, etc.)

- Along with the number, please provide a short label for each problem. E.g., "Algorithm for computing $\sqrt{n}$," "Algorithm to find common elements in sorted lists," "Locker doors puzzle," "'Asymptotic notation practice with limit definitions," etc.

- Problems must appear in order. If you cannot complete a given problem, write the problem number and label, and indicate that you do not have a solution. Otherwise you may waste the grader's time looking for the problem.

- Leave adequate spacing. One page per problem is preferred, but anything that clearly indicates problem boundaries is acceptable.

- When submitting electronically, take good scans and/or photos and upload as a single PDF file.

- To specify an algorithm, any reasonable and consistent psuedocode is acceptable.

- At the end of the book (just before the index), you will find hints for most of the problems. I suggest that you first try to do each problem without the hint, but if you get stuck, feel free to use the hint.

- I encourage you to discuss homework problems with classmates and with me, but all work that you write up should be **your own and in your own words**. You may make free use of the textbook and class notes, but you must **cite any other source you use** (website, etc.)

Problems

1. (Levitin 1.1.4, modified) Design an algorithm for computing $\lfloor \sqrt{n} \rfloor$ given any positive integer n. Besides assignment and comparison, your algorithm may only use the four basic arithmetical operations. For your answer, describe your algorithm in precise pseudocode. Note that there is no efficiency requirement.