CSCI 3104: Algorithms
Homework 1

Due at **11:00am on Wednesday, September 9, 2015**. Submit electronically at moodle (name file as **LastName_FirstName_HW1.pdf**) or submit in paper before class. Make sure to include your name and student ID. Digital submission should also include the Honor Code Pledge (`http://honorcode.colorado.edu/about-honor-code`), and paper submission should include your signature indicating adherence to the Honor Code Pledge .

1. Algorithms are designed to solve certain computation tasks that are specified by the expected input and output. For each of the following problems, how would you specify its input and output? (a) driving direction; (b) course scheduling; (c) Google web search engine; (d) Amazon item recommendation; (e) Fedex shipment scheduling.

2. Find an online recipe for one of your favorite dishes. Provide the URL and a copy of the recipe. Are the instructions precise and unambiguous as algorithms? Briefly explain why.

3. In each of the following situations, indicate whether $f = O(g)$, or $f = \Omega(g)$, or both (in which case $f = \Theta(g)$). Briefly explain why.

   (a) $f(n) = 32n^7 + 48n^3$ $\quad$ $g(n) = 50n^5 + 40n^4 + 100$
   (b) $f(n) = \log(8n^2)$ $\quad$ $g(n) = \log(n^3)$
   (c) $f(n) = 3^n/7 + n^3$ $\quad$ $g(n) = n^5 \log n$
   (d) $f(n) = (\frac{5}{3})^n$ $\quad$ $g(n) = 8n^8 + 63 \log n$
   (e) $f(n) = (n-1)!$ $\quad$ $g(n) = n!$

4. Write a python program that (1) takes one argument n as input, (2) generates n random integers in the range of $[0, 10n]$ and sorts the n integers in ascending order using the selection sort algorithm introduced in Lecture 1, and (3) repeats step (2) for 10 times and prints out the running time (min, max, and average) across the 10 rounds of sorting.

   Name your python code as **LastName_FirstName_HW1.py** and submit electronically at moodle. In addition, test your code with $n = 10^3, 10^4, 10^5, 10^6$ and report the running time for each n value. If the program takes too long to run for bigger n, you do not have to finish it and report the actual running time. Instead, include a statement estimating the running time based on the running time for smaller n values and the big-O time complexity analysis. Provide your answers along with your source code, and specify the type of machine you used for testing (CPU, memory size, OS).