605.621 Foundations of Algorithms Spring 2025
Programming Assignment #3
Assigned with Module 8, Due at the end of Module 10 (April 7)

The goals of Programming Assignment 3 are: (1) to practice your algorithm design and specification, (2) to practice your trace runs and tests to measure asymptotic behavior, (3) to practice analysis of theoretical asymptotic behavior compared to actual program asymptotic behavior, and (4) to exercise the algorithm analysis techniques you have studied.

This programming assignment directs you to make specific changes to specific algorithms in the course text. All page references are to CLRS hardback/online books. **Do not** use or modify the algorithms in *Problem 7-1 Hoare partition correctness* (page 199/195), *Problem 7-6 Median-of-3 partition* (page 203/199). There are various median-of-three partitioning algorithms and programs on the Internet not consistent with the instructions in this assignment. Solutions that do not follow the specific instructions in this assignment will receive a score of zero on that part of the solution.

In this course, critical thinking and problem analysis results in discovering appropriate, and possible better or best, algorithm for a problem; defining the algorithm in pseudocode; demonstrating (we don't usually prove) that the algorithm is correct; and determining the asymptotic runtime for the algorithm using one or more of the tools for asymptotic analysis you have studied this semester. This programming problem is not a collaborative assignment.

Each algorithm you design must follow the **Programming Assignment Guidelines & Pseudocode Restrictions** on Canvas page *Programming Assignment Guidelines including Pseudocode Restrictions*)when preparing your solutions to these problems.

Exercising the algorithm you design using one or more example data set(s) removes any doubt from the grader that the algorithm is structurally correct and all computations are correct. When a problem supplies data you are expected to use it to demonstrate that your algorithm is correct.

You are permitted to use Internet resources while solving problems, however complete references must be provided. Please follow the Sheridan Libraries' citation guidance at "*Citing Other Things* - HOW DO YOU CITE AN INTERVIEW, A TWEET, OR A PUBLIC WEB PAGE?" and continue to the APA Academic Writer *Sample References*. Additional example citations are provided at the Purdue University, Purdue Online Writing Lab (OWL), College of Liberal Arts *Reference List: Electronic Sources*. You can also use training provided by Victoria University Library, Melbourne Australia on *Oxford Referencing: Internet/websites*

1. Given an array , $a[i], ..., a[j]$, with $j - i \geq 2$, let $k = \lfloor (i + j)/2 \rfloor$, choose the median value amongst $a[i]$, $a[j]$, and $a[k]$ (i.e., the middle value if $a[i]$, $a[j]$, and $a[k]$ were sorted) as the partition element for QUICKSORT(A, p, r) pg. 183/179. We will call this median-of-three partitioning.

   (a) [5 points] Write pseudocode for a modification of PARTITION(A, p, r), pg. 184/180, using median-of-three partitioning.

   (b) [10 points] What is the worst-case asymptotic behavior of QUICKSORT(A, p, r) using median-of-three partitioning? Provide proofs supporting your conclusions.

   (c) [10 points] What is the worst-case asymptotic behavior of QUICKSORT(A, p, r) using median-of-three partitioning on an input set that is already sorted? Provide a proof supporting your conclusion.

   (d) [20 points] Implement QUICKSORT(A, p, r), using:

      i. [10 points] PARTITION(A, p, r) on page 184/180, and
      ii. [10 points] PARTITION(A, p, r) using median-of-three partitioning.

   (e) [20 points] Test your implementations to verify analytically derived complexity bounds or bounds developed on pgs 183-190:

      i. [10 points] Worst-case metrics for the implementation of QUICKSORT(A, p, r) with PARTITION(A, p, r) on page 184/180,
      ii. 10 points] Worst-case metrics for the implementation of QUICKSORT(A, p, r) with PARTITION(A, p, r) using median-of-three partitioning.

   (f) [15 points] Attributes of your code:

      i. [5 points] Code follows a reasonable, consistent style with reasonable documentation, with appropriate use of structures, modularity, error checking
      ii. [5 points] Trace runs documenting that your program executes correctly.
      iii. [5 points] Instrumentation and tests to measure the asymptotic behavior of your program.

   (g) [20 points] Analysis comparing your algorithm's worst-case asymptotic behavior to the achieved asymptotic behavior of your programs [1(b), 1(e)(ii)] .