

# CSCI 230 -- Lab 5

## Selection Problem

Due: \_\_\_\_\_

---

Feel free to discuss and help each other out, but does not imply that you can give away your code or your answers! Make sure to read all instructions before attempting this lab. You can work with a lab partner and submit one lab package for your group.

**You must use an appropriate provided template from Canvas or my website ([zeus.mtsac.edu/~tvo](http://zeus.mtsac.edu/~tvo)) and output "Author: Your Name(s)" for all your programs. If you are modifying an existing program, use "Modified by: Your Name(s)".**

**Lab question 1:** Provide some good reasons that you might not want to sort the data in order to determine the median value.

**Lab question 2:** Explain how the prune-and search design pattern works. Is recursion always necessary? Explain.

**Exercise 1:** Modify selection sort to find the  $k^{\text{th}}$  smallest element (do not sort the whole list). Test it with a small file with a few integers first like  $n = 10$  (try  $k = 1$ ,  $k = n/2$ , and  $k = n$ ) and then try it to find the median of the smaller data file from project 2, *smallRan1k.txt*.

**Exercise 2:** Implement the randomized quick select algorithm to find  $k^{\text{th}}$  smallest element as discussed in lecture/book. Test it with a small file with a few integers first like  $n = 10$  (try  $k = 1$ ,  $k = n/2$ , and  $k = n$ ) and then try it to find the medians of the two data files from project 2, *smallRan1k.txt* and *large100k.txt*.

**Extra Credit:** Determine the maximum depth of recursive calls (not number of recursive calls) for exercise 2 and print it out. Confirm that it should be  $O(\log n)$ .

**Online Submission:** Submit one PDF file via Canvas includes status, answers to lab questions, output and source code for all required programs.