Programming 2
Cptr.212

Fall 2020 | Southwestern College
Professor Evan Daniel
Evan.Daniel@SCKans.edu

Assignment #11
"Merge Sort and Quick Sort"

DELIVERABLES

Revisit the Leetcode problem **Sort an Array**, and complete the problem using the Quick Sort and Merge Sort algorithms. On Friday, October 23rd, I will meet with each student and they will complete the problem using one of the two in 15 minutes while sharing their screen on Zoom. Students are expected to provide an explanation of what they're doing while they're completing the problem.

I will also check on Friday that you are able to run the sample Win32 API code on your local machine. As per our class discussion, that will require you to install **Git for Windows** (includes Git Bash) and **MinGW** (Minimalist GNU for Windows).

COMMENTS

Quick Sort and Merge Sort are on the list of classic algorithms in almost any CS context. We will discuss in class why they are effective, and when each is more appropriate than alternatives.

As we begin discussing the use of C towards software development — and how the techniques we have discussed relate to the production of applications — we will need to add some new tools to our toolbox. MinGW will allow us to use GCC and the headers we need, including those we already expect to be there (such as <stdio.h>) as well as <windows.h>.

RESOURCES

For information on Quick Sort and Merge Sort, see the discussion of the Leetcode problem; the Wikipedia articles on **Quick Sort** and **Merge Sort** also happen to be quite informative.

While we are only installing prerequisites for the Win32 API this week, it would be helpful to read the first section or two of **theForger's Win32 API Programming Tutorial**.