Devin Gendron
Assignment 7 Comparisons (Group 30)
11/15/17

Adrian Pangelinan Comparison –

For the findMedian assignment, Adrian and I had similar structure to our assignments. Our functions obviously had the same parameters due to the assignment specifications and were sorted by the method in the notes. We both used an if else function to determine the parity of the array size in order to begin finding the median of it. I believe that our code is fairly even in quality, but if I had to choose where I did better, I would say that the readability of my program is better. I blocked up the sections of my code and made it clear what each section did. Adrian's attempted to do this, but having return methods outside the function scope makes the reader have to jump around rather than read straight through the code. Where I believe Adrian's code is better is in their use of variable naming. Using names such as, medianIndex, medianIndexLow, and medianIndexHigh, are more straight forward and easier to understand due to their mathematical nature. I used names that made sense to me, but might not be so for other people. I thought his variables were named in a way that anyone could read the program and understand.

For the stdDev code, I think my code was better by fitting more of the required code into the for-loops. By doing this, I believe that my code had a stronger sense of structure and quality by shortening the process into less steps. I believe Adrian did a better job at commenting out the process. By defining processes like, "For each element of array, calculate the square difference = (age – average_age)^2 and add it to the summation", Adrian created a stronger sense of understanding by clearly defining the procedures of the code in his comments.

Christina Everdyke Comparison –

In comparison with the findMedian program, I believe my program was better at portraying how the median was found. I broke up each step to make the program more incremental while still keeping the code concise. Christina's code, to me, isn't as clear - although it is shorter. This is where I believe she did better. Her code's readability and short concise style makes for compact, well written, aesthetic code.

In comparison with the stdDev program, I believe my program was better at compartmentalizing the code into the for-loops. Again, I broke the math into a more incremental visual process in order to portray the steps that were being taken. This could be seen as a double edge sword, because it increases readability, but makes the code longer. I believe Christina's code looks much cleaner than mine. It's compact and flows just like her findMedian program. She strikes a good balance of what steps to include in the code and its readability.

Jared Sulzdorf Comparison –

In comparison with Jared's findMedian program, I found that my structure was a little straight forward because of Jared's style of commenting large chunks of code. I believe my comments were better because of this. There seems to be a fine line where too much commenting between lines can make the readability of the code difficult. Where too little could leave someone reading your code lost. I believe that if Jared made his code's comments more succinct and spread out with each function or execution, then his code would be much cleaner and easier to understand at first glance. Where I believe Jared did better was in compacting his code's steps into fewer lines. I believe that by my breaking the process up, it makes it easier to understand, but fitting three lines of code onto one looks better in the long run.

In comparison with Jared's stdDev program, I had a difficult time reading his code due to the formatting (perhaps just when converted to a text file). Thus, I believe my code is better in its presentation and its structure. I can clearly read my code and understand it by its incremental breakdown of the necessary steps to find the standard deviation. I'm feeling lost in Jared's, trying to decipher what is hard coded and what is just a comment. I believe that although the implementation of Jared's comments is off, the information within the comments are valuable and do a better job of presenting what the code is doing, compared to my code. For example Jared says, "We need to compute the average age of all the Person ages in the Person array before doing anything else. So we loop over each value in the array and add it to an accumulator", whereas I say, "for loop for mean". I do specify each line of code as I write the for loop, but Jared's comments are clearly more in depth than mine and create a better narrative of what the code is doing.

From looking at my groups code, I've learned that there is much more room to be concise with my code as well as more in depth with my comments. The aesthetic style is an important factor to working code that shouldn't be overlooked. I believe that there is a balance that should be maintained when writing comments and code – that is, comments should be in depth, but not overly long unless doing a large comment block. Or that concise code can function just as well as longer code, but then readability or understanding could be lost when steps aren't as explicit. My goal from this point will be to write better comments in my code. I'd like to present a clearer picture as my classmates have done. I'll also be focusing on making more code more concise and aesthetic. Christina and Adrian did an excellent job in writing their code – I found their presentation to be excellent and I hope to make mine just as good or better in future assignments.