Project Plan 3A Group Assessment
Students: Wan Ashraf Wan Ahmad Ezani, Jacob Carter, Emily Edmondson, Chad Fingerson

- Which project plan did your group decide on?

We decided to combine two different project plans. We chose to use the flowchart from Wan's Project Plan and the Tests from Jacob's Project Plan. This allows us to borrow the strengths of the two different plans.

- What advantages do you think that project plan has over the others? (be detailed)

Wan's Project Plan Code Design:

Wan's Project Plan for Assignment 3A used a flowchart for his design. Our group thought this was beneficial due to the increased visibility and legibility the flow chart offers compared to the pseudocode our other 3 members wrote. Overall it gave the project a higher level of readability. Specifically, Wan's Project Plan Flowchart elegantly displays the logic behind a nested loop when number is tested against min and max and then looped back to the beginning test against the counter. The pseudocode demonstrates this as well, but not with the legibility that Wan's flowchart offers. The pseudocode demonstrations of nested loops depend heavily on the reader's prior understanding of nested loops in C++ and the flowchart has the advantage of displaying these nested loops in a way that is more accessible to any reader.

Following the same logic as above, Wan's flowchart has the advantage of displaying how a counter works to a reader. In Wan's flowchart the counter incrementation occurs in conjunction with the flowchart arrow bringing the reader back to the beginning of the loop for the counter-based condition test. This makes it clear exactly what aspects of the program counter is controlling and the logic of its implementation.

Ultimately, the excellent legibility of the flowchart provides a framework for comprehension, guidance, and duplication in any future code project plans an individual might be planning to write.

Jacob's Project Plan Test Cases:

Jacob tested more edge cases than the rest of our group. He made sure to check large values for input and in addition was the only member to show testing a high amount of inputs as well.

Additionally, Jacob's test cases were broken into two separate sections, one for each of the prompts that the user would receive. This allows for a more comprehensive view of the individual steps of the program and could be beneficial in the event that the program contains a bug.

- What improvements do you think could be made to that project plan? (be detailed)

Wan's Project Plan Code Design:

A logic issue did arise in Wan's project plan where min and max are initialized =0. (Incidentally, only in Chad Fingerson's Project Plan did some version of this where the error did not occur.). The problem setting a predefined maximum and minimum is we are assuming what the user is going to use as values. Since we have no idea what values the user plans on using we are not able to safely make that assumption. Instead the plan should have used the first number from the user as the minimum and maximum value.

In addition Wan's flowchart does not show incrementing the counter after setting the min and max value. Overall the logic of the loop in the flowchart was solid but did not show the continuation after setting the max and min values.

The last issue with the flow chart as designed is the input for user values is outside of the loop. In order for the loop to work correctly the input would have to be inside the loop. It would be okay to accept the first input from the user outside the loop in order to set the initial maximum and minimum value but every subsequent input should be inside the loop. This guarantees we get all the necessary inputs and they are all checked for new minimum and maximum values.

Jacob's Test Cases:

Jacob's test cases tested far more edge cases than the rest of our group. Although he did not test as many special cases for user input. Adding in situations where the user enters numbers randomly was tested but no numbers from low to high and high to low. If those edge cases are not tested we could potentially have an error in our logic when we assign our maximum and minimum values.

One specific case that Jacob's test cases did not take into account is a situation in which the user enters only negative numbers. This leaves a hole in the logical testing of the program, and if taken into consideration with the min/max values from Wan's flowchart, would likely lead to false confidence that the program would work as intended. Thus, the test cases would be improved by the inclusion of a test of only negative numbers.