REVIEW FEEDBACK

Jason Wong 24/03

24 March 2020 / 10:00 AM / Reviewer:

Steady – You credibly demonstrated this in the session.
Improving – You did not credibly demonstrate this yet.

GENERAL FEEDBACK

Feedback: This was a great session, There were only two issues here 1 was that you didn't ask about edge cases. The other was that you had a small issue with debugging at one point in the session. Looking forward to the next session.

I CAN TDD ANYTHING – Steady

Feedback: I liked that you started out with a simple test such as ([45], 30. 50) expecting [45] which was strictly based on the operating conditions of the program. You then incrementally increased the complexity with the tests ([25], 30. 50) expecting [30] and ([55], 30. 50) expecting [50]. This method of testing displayed a good understanding of how to develop user tests in an incremental, evolutionary way.

I CAN PROGRAM FLUENTLY – Steady

Feedback: I liked that you initialised git before proceeding with any development. You have a good grasp of core programming concepts such as OOP, if statements, variables and loops. I really liked that you were using the input output table as a backlog of features to be built and maintaining this throughout. You have a good understanding of basic and intermediate Ruby syntax and generally seem comfortable in the language.

I CAN DEBUG ANYTHING – Improving

Feedback:

I CAN MODEL ANYTHING – Steady

Feedback: You modelled your solution in a single method which I felt was a nice and simple implementation and provided a good place to start.

I CAN REFACTOR ANYTHING –Steady

Feedback: I felt that you followed a solid Red-Green-Refactor cycle where you were making sure to write good tests that were some evolution of the previous tests. You were then moving onto the Green step where you were solving the test in the most efficient and simplistic way you could. Then looking for opportunities for refactoring after you were hitting Green for your tests, and making refactors to your code where appropriate.

I HAVE A METHODICAL APPROACH TO SOLVING PROBLEMS – Steady

Feedback: Overall you seemed to have a pretty good methodical process during the session where you were prioritising tasks that provided immediate value to the user. I liked that you were prioritising acceptance criteria over edge cases. You worked through tests in a logical manner starting in the most simple of spaces and moving towards more complex spaces. Your cycle is pretty solid and regular.

I USE AN AGILE DEVELOPMENT PROCESS – Improving

Feedback: Your information gathering was good. You asked some good questions about the input/output. You asked some questions about formatting and data types. I liked that you provided a set of inputs and outputs as acceptance criteria for the program. I would have liked you to ask some questions about edge cases right at the beginning, you did end up considering some edge cases right at the end.

I WRITE CODE THAT IS EASY TO CHANGE – Steady

Feedback: Overall you write code that is easy to change and maintain, you had your test suite properly decoupled from your implementation by making sure the tests are based solely on behaviours this makes changes to the code much easier. I liked your use of git. You committed after every passing test which created a good paper trail for yourself and developers coming after you.

I CAN JUSTIFY THE WAY I WORK – Steady

Feedback: Throughout the session, you were vocalising your thought process. As a result, I was able to clearly follow you through the session. You seemed to be able to justify your decisions for doing things. You were regularly updating the product owner on your progress through the task.