Reflection

Assignment 7 – Exercise 18.2 – String Processor

My implementation of secretCode was causing an error where some offset was causing the output String to be incorrect. I used a unit test to test different offsets which are above and below 26. The tests below the offset of 25 worked fine. This is when I realized I was using a modulo of 25 instead of 26 since there are 26 alphabets whenever the offset was bigger than 25 the incorrect modulo would cause the alphabet to wrap around incorrectly skipping one alphabet. After setting the modulo to 26 the output of the unit tests came out correctly.

I did not test for negative offsets last time and I wanted to be sure for this submission, so I wrote a unit tests for negative offsets. I was surprised to see them come out negative. The only possible point of error I could see was the modulus operator. So, I tested the following in java -5%10 which outputted -5 where I was expecting 5. To solve this issue, I wrote another method which would check if the offset were negative and if so, it would add 26 to the output of the modulo equation. Then I ran the unit tests again and it came out positive.

Assignment 3 – Exercise 8.5 – Eight Queens

When I wrote the setQueen method I overlooked that it must return if it can or cannot set the queen in that spot as a Boolean value. This was a simple fix. After adding this new functionality, I realized that I could make the setQueens method simpler and more efficient by using the new setQueen method. Integrating this new element needed me to remove most of the previous implementation since some of it was redundant and not useful. The new implementation was much smaller due to the modularity. I tested the functionality with unit test cases like placing more than 8 queens and placing few queens over few function calls. I also tested setQueen to make sure that was not causing any unexpected errors since it was at the root of setQueens method.