Problem 3:

Let's say that for example I wanted to take a test-driven approach to a login-credentials authentication system. The first thing I would do is set up a set of inputs that I would want to pass the authentication system and in a test. This can be broken down into multiple tests as well. I can also write tests for sets of inputs that don't get verified.

Let's say that the first set of inputs are tests for illegal characters. I would write a test that verifies if a program detects an illegal character, then after the test fails I would then write a part of the component that checks for certain characters in a string.

After that, we can then write a test to check if the component verifies that the password is within a certain length, fail the test, then write code to pass it.

Essentially, we increment through the desired functionality through our test and constantly add or modify our code so that all of the tests pass. Changes can be easily tracked using version control.