CS 362 - Software Engineering II
Homework 7

For the following programming problems you will need to follow the instructions below:

1. For each programming problem you will follow the steps of Test-driven development. For each problem you will have two files.
1. 1 File that contains the tests.

1. 2 File that contains the code/implementation.

2. You will commit every time you write a test and write the code that makes your test pass. If you can remember the git demo lecture, commits have commit messages. Each commit needs to have a commit message that explains what the commit was for.

Here, you will have a commit for the test and its commit message, and a commit for the code that passes the test and its commit message.

Note - You will lose points if you do not commit after every test and lose points if you do not commit after writing code that passes that particular test. You will lose points if you do not add a commit message.

Steps and Deliverables

- Read program specification
- Write initial test and run it to ensure it fails
- Commit changes, Example: "Wrote first test, {test description}”
- Write code just enough to pass the test
- Run a test to verify if it passed
- Commit changes, "Implemented x method {implementation description}"
- Write second test

- Repeat

Deliverable

Submit files that contain code and the tests to github. Your submission will be a link to a github repository.

Question 1 **15 points**

FizzBuzz
Specification: Write a program that prints the numbers from 1 to 100. But for multiples of three print “Fizz” instead of the number and for the multiples of five print “Buzz”. For numbers which are multiples of both three and five print “FizzBuzz”.

Deliverable:
Follow the instructions and submit your code and tests to github.

Question 2 **15 points**

Leap Year
Specification: From homework 1, rewrite the leap year program using the test-first approach.

Conditions for a leap year:

> “Years that are evenly divisible by 4 except
> years that are evenly divisible by 100
> unless the years are also evenly divisible by 400”.

Deliverable:
Follow the instructions and submit your code and tests to github.

Question 3 **10 Points**

How would you apply test driven development within your social media system from in-class activity 1? Explain with a scenario.

Deliverable:
Select **one** part of your social media system and explain how you would develop it using the test-first approach.

Example:

If you were working on the login functionality of your social media system, your TDD approach for it would be:

- Create a test that redirects when a user is not logged in.
- Create code to check if the user is already logged in, or is logging in (authenticate).
- Create code to redirect to the login page.
- Create a test that calls the above, verifies the login works ok.
- Create the login page itself.
