Session 9 - Testing

1 Testing questions

1. Explain, in your own words, the concept of exceptions

2. Explain, in your own words, the concept of code coverage

3. Explain, in your own words, the process of test-driven development

2 Circle

1. Write a class to represent a circle. A circle has a center and a radius.

   - Add a method to determine if a point (x, y) is inside the circle.
   - Add a method that takes another circle as input and returns whether the two circles overlap
   - Add tests for these methods

3 BadList

1. Write tests and correct the bad code in BadList.cs: `https://gist.github.com/boegholm/bb2c11f343b24ab121d84f1daaa4fd56`..

4 Vector

1. Write a class to represent a 2D vector.

   - Add an appropriate constructor and methods for addition and subtraction of vectors.
   - Add methods to compute the scalar and cross product.
   - Add tests for all methods

5 String

1. Write your own String class. Internally the class should use an array of characters to represent the string.

   - Add the following methods: CharAt, length, substring, ToLowerCase, and Equals. You must not use any part of the .NET string type in your implementation.
   - Create tests for the methods

6 Time Duration

1. Write a class to represent a time duration. Internally the class should store the time in miliseconds.

   - The class should expose two constructors:
     - Duration(long miliseconds)
     - Duration(long hours, long minutes, long seconds)
   - Add methods for Adding and subtracting to the duration. These methods should take a time duration as input.
   - Add a method for multiple the time by an integer value
   - Add test for all methods

7 IntegerStack

1. Write a class to represent a stack of integers. Internally the class should use an array of integers to represent the stack.

   - Add methods for pushing and popping elements of the stack.
   - Add tests