**Custom List Class Project (out of 125 points)**

**User Stories**

The built-in List<T> class is a generic class that acts as a wrapper over the array class. **You cannot use built-in List or Array methods.**

**(20 points)** As a developer, I want to use Test Driven Development (TDD), so that I can write tests for my methods to pass to ensure proper functionality within my application. There needs to be several tests per method.

**(5 points):** As a developer, I want to make good, consistent commits.

**(10 points):** As a developer, I want to use a custom-built list class that stores its values in an array, so that I can store any data type in my collection.

**(10 points):** As a developer, I want the ability to add an object to an instance of my custom-built list class.

**(10 points):** As a developer, I want the ability to remove an object from an instance of my custom-built list class.

**(10 points):** As a developer, I want the custom list class to be iterable.

**(10 points):** As a developer, I want to be able to override the ToString method that converts the contents of the custom list to a string.

**(10 points):** As a developer, I want to be able to overload the + operator, so that I can add two instances of the custom list class together.

**(10 points):** As a developer, I want to be able to overload the – operator, so that I can subtract one instance of a custom list class from another instance of a custom list class.

**(10 points):** As a developer, I want a Count property implemented on the custom-built list class, so that I can get a count of the number of elements in my custom list class instance.

**(10 points):** As a developer, I want the ability to zip two custom list class instances together in the form of a zipper. An example:

* I have List<int> odd = new List<int>() {1,3,5}; and List<int> even = new List<int>() {2,4,6};
* When lists odd and even are zipped together, your new list will contain values 1,2,3,4,5,6

**(10 points):** As a developer, I want to use C# best practices, SOLID design principles, and good naming conventions on the project.

**(Bonus 5 points):** As a developer, I want the ability to sort an instance of my custom-built list class. To be eligible for the bonus points, you may not use Array.Sort() that is already built in and you must tell us what sorting algorithm you used.

**NOTICE: get your unit tests (test methods) checked off by an instructor before you begin writing your methods to ensure you are on the correct path.**