WEEK 12 & 13

1. Preparation for Assignment

If, and *only if* you can truthfully assert the truthfulness of each comprehension and memory self-check statement below are you ready to start the tasks for the week.

1.1. **Reading Comprehension Self-Check.**

- I know and understand the difference between an array, a vector, a stack, a queue, a deque, a set, a list, a tree, a BST, a map, and a graph.
- I know and understand that nodes and vertices are the same thing.
- I know and understand how and why edges connect vertices.
- I understand why the 'data' for a graph is the set of vertices and all of the edges.

1.2. **Memory Self-Check.** I can, and have, explained to someone who is not a student in the Computer Science and Electrical Engineering, Computer Information Technology, or Mathematics departments in a way that they understand what a graph is and why it is important.

2. Week 12 & 13 Team Tasks

Note: All review tasks come from the book.

2.1. **Review Task 13.1.**

2.2. **Review Task 13.3.**

2.3. **Review Task 13.4.**

2.4. **Creation Tasks.**

(1) Write stub classes for one of the UML class diagrams on pages 363 and 364 of the textbook.
(2) Write a complete but "first draft" of a unit test for your graph class.
(3) Fill in the methods for you class to match your unit test. At the same time, fix any defects you find in your unit test.

2.5. **Pondering Task.** What different kinds of uses can you find or brainstorm for this data structure?

Date: January 7, 2020.

2.6. **Pondering Task.** What changes to the API for this data structure can you brainstorm that might make it easier to use? What would be the design for one of your changes?

2.7. **Pondering Task.** Brainstorm some coding tasks/situations or types of data where using this data structure would make organizing, accessing, inserting, or updating the data difficult.

2.8. **Pondering Task.** Brainstorm about and record why there are differences between the API for this data structure and the API for this data structure in another language of your choice. Consider the strengths, weaknesses, similarities, and differences between these API's for the data structure.