* Overview
  + Overall I feel like I learned a lot in this class. I took CS 1400 last semester which means that I am taking CS 1440 at the recommended time for CS majors. I think it is important to have and understand a tool like UML to be able to have an effective precoding process so that when it comes time to code you are able to. However, I felt like we got to complicated to fast. I would have like to learn more basic material that seemed to just be brushed over and expected for us to figure out on our own. For example, writing good test cases and understanding how and what to test or even more basic, getting and checking valid user input. This isn’t something we spent time talking about and it is something I feel is important to learn and this would be the class to learn that skill.
  + If I were to restructure the class it would start with UML and talking more about the precoding process. Different methods of figuring out what you need. I have found even before making a UML diagram it is more effective for me to sit down and drawn it out on my own on a white board and write in my own words what the program is supposed to do the identify classes and write down specifically in my own words what that class is supposed to do. Then, after how to have an effective prewriting process is understood move into design decisions and how to write maintainable code. Then to extendable code. Then reusable.
  + We talked a lot about how in a software developing career we will often be put into teams. Some of these projects seemed very overwhelming as a fresh out of 1400 coding student. I think it would be really fun and helpful to have a semi-complex project that we were broken into groups and we each had to write a class and the test cases for it. Then, we had to coordinate with one another and use GIT to share code.
* Assignments
  + Assignment 1
    - Great assignment I thought it was a perfect introductory assignment
  + Assignment 2
    - This assignment was a ways beyond the level of the class.
  + Assignment 3
    - Good assignment, I was a bit confused on what uncovering a bug meant I thought I had gotten so many but apparently I didn’t
  + Assignment 4
    - The Bingo assignment was one of my favorite assignments. I just had a really hard time knowing what and how to test the bingo cards. I would suggest adding (at least just for this one assignment) specific things to test for. Such as validating user input and talk in class how to do so.
  + Assignment 5
    - This assignment I felt was just slightly out of range for me. A big concept in the HW was recursion and we were just learning how to use recursion toward the end of this assignment. The starter code was also very complex and difficult to understand. I would suggest possibly using GeoRegions as the in class example and having the farm animal example be the homework assignment. Which would include a basic farm animal pure virtual class.
  + Assignment 6
    - I thought the dictionary assignment was a lot of fun. It was a bit of a pain to get my head wrapped around the concept of generics and with all the test cases factored in I felt like it pushed me just to the edge but not over
  + Assignment 7
    - This assignment was good and I felt like it was doable the actual algorithms were difficult to figure out but the concepts weren’t overly difficult. A little more assistance with the design would have been appreciated.
* Quizzes
  + Quizzes were great, I felt like in taking the quizzes is where I was able to put a lot of things together. The one thing I would suggest would be open the quizzes up to 2 or 3 attempts. I missed one question worth 5 points on a quiz and it dropped my grade a full percent. An alternative would be set the total quiz grade weight to be a set percentage of the final grade.
* Tests
  + Tests were good but a lot of the time the questions were worded weird and hard to understand.