For this Lab I did what I did in Lab 2/3 and stuck to designing the members of the classes as close to the description as possible. This lab, since we also had example output I also designed my classes to match the given examples without adding anything that would show up on a class diagram.

One of the challenges I faced was when I was printing out the contents of the CourseManager and I got an error relating to the EvaluationText property in the Evaluations.cs file. Through the use of breakpoints and other debugging tools in Visual Studio I managed to figure out that the error was because some Evaluations in the course had a null textFile value which was causing errors in the logic of the EvaluationText property getter.

I also once again learned the importance of carefully reviewing class diagrams and instructions multiple times, as I had forgotten to implement some parts of the class in my first initial programming sessions. An example being the Assignment class where when I first implemented it I had forgotten to add functionality to the AddTask method that would set the task due date to the same due date of the assignment. It was only when I was reviewing my output and comparing it to the example output that I realized I was missing that part.