1. **Names**
    a. Kevin Ash and Parker Banks
2. **Final Statement**
    a. Overall, most of what we wanted to implement was implemented successfully. We are happy with the way that the UI looks and the customization options we have available. We were planning to include more customization, but a lot of it would not have been very useful and would have messed with finely tuned parameters that likely would have resulted in an uninteresting simulation (like all of the predators and prey dying before they were able to even get near each other). We successfully implemented a neural network to control the animals, but the evolution system is not as robust as we would have liked. The offspring inherit their parents neural networks, but we unfortunately did not implement random mutations to these networks.
3. **Final Class Diagram and Comparison**
    a. The final class diagram can be found in the GitHub repository (because if pictures are included they are too low quality to read)
    b. The patterns that we used in designing this project were the observer pattern, factory pattern, singleton pattern, and the use of abstract classes.
    c. The main changes from the class diagram for projects 5 and 6 were the addition of the patterns. For projects 5 and 6 we were just using the Singleton pattern with a plan to add the factory pattern, but for project 7 we have added observer and factory. There were also some changes to the fields and functions within the classes as we discovered we needed more functionality or a new helper funciton should be created.
4. **Third-party Code vs. Original Code Statment**
    a. The neural network code was taken from a GitHub repository that I created a long time ago based on this video: https://www.youtube.com/watch?v=sK9AbJ4P8ao. This code was adapted to work in C# and then adapted even further to fit our specific use case. All other code was original code.
5. **Statement on the OOAD Process**
    a. One of the hardest parts of the design process for us was deciding on a project we wanted to do. We had a lot of ideas but many of them were either too simple or too complex (and arguably the project we settled on is still too complex). We also had trouble deciding what framework we were going to use for the UI. After a little research we started with Windows Forms, but we quickly realized that it was outdated and switch to WPF instead. As for designing the code itself, we struggled to come up with a way to handle storing the map, and we are still not totally happy with the way we did it. You have to pass the map object to many different functions that shouldn't really depend on it, but it was the best solution we could come up with.