**Presentation Notes**

Intro 🡪 Nam  
Our Client 🡪 Thomas  
Our Task 🡪 Ana

Breakdown

* Interactive Selection 🡪 Nam
* Drawing of the Organisms 🡪 Thomas
* Saving/Loading Organisms 🡪 Ana
* Displaying Succession Tree 🡪 Nam

Issues

* Insert Your Own Issue??? 🡪 Thomas
* Screen Class (the old screen class) 🡪 Ana
* Insert Your Own Issue??? 🡪 Nam

Societal/School Benefits

* Client Benefit 🡪 Ana, Thomas, and Nam

Environmental Impact

* Paperless 🡪 Thomas
* Electricity Use 🡪 Nam
* Education 🡪 Ana

Demo

* Demo New Organism and Reproduce (also mention keyboard shortcuts but only for reproduce and selecting the children, and zooming) 🡪 Thomas
* Demo View Genes, Animation, and Help (also mention any shortcuts that have to do with these) 🡪 Nam
* Demo Create Your Own Organism! and Saving/Loading 🡪 Ana

**Script**

Ok, so our task! The problem we were given by our client to make into an application was a rather unique one. So from a book titled *The Blind Watchmaker* by Richard Dawkins the idea for a computer program that would create organisms and allow them to evolve over generations was born. Our client gave us a copy of a passage from this book and our task was to essentially bring this program to life and add a few extra features that will be explained as we continue the presentation. So in its simplest we had to graphically display organisms whose appearances were influenced by a specific set of genes that we coded and we had to allow the organism to reproduce and create children with slight variances in the genes for the user to select which one to survive. And that was our basic task, but of course we added awesome bonus things to make it more interesting and versatile for our client.

So, another feature of our program is the saving and loading of organisms. Now in *The Blind Watchmaker*, Dawkins mentioned how he once stumbled upon a really cool looking organism but he had no way to save it and sadly lost it forever. He tried hard to recreate it but he just couldn’t. So you can see how saving and loading would be a very important part of our program. So the saving component allows the user to enter the name of their organism as the name of a text file that they can save anywhere on their computer. It saves not only the genes belonging to the organism that the user is trying to save but also the genes of every single one of its ancestors, and you will find out why when Nam tells you about our next cool feature. And loading works by letting the user select the file where the previously saved an organism and then displaying this organism onto the screen.

Another issue that we ran into was when setting up the layout of the different screens for our application. So in the earlier stages of our program we had made a Screen class which was supposed to set up everything that would be displayed on the screen and then our main class would simply add it to the main frame. But this became very problematic because we were using graphics in that screen class to draw the organisms inside JPanels and then add these JPanels into the screen JPanel to then be returned to the main class. This class also had all sort of different methods for the different screens and I thought it was just getting too clunky and it had to be simplified. So to fix this my idea was to create a new class called Drawing that would extend a JLabel and in that label draw a single organism. Now these JLabels would be added to a JPanel inside another of our classes called Evolution and finally this JPanel was added to the main frame. Although it might be a little hard to understand this without seeing it this solution made it much easier to understand what each bit of code was doing and made our program more in line with Object Oriented Programming.

Ok, so school benefits. So obviously our program is mainly beneficial for our client, as it allows her to have a good tool to sort of visually explain the processes of evolution to students. But our program doesn’t necessarily have to be only used by our client, other biology teachers in the school and even other school could use as a tool to demonstrate the ideas behind evolution and genetics to students. Our program would be a good tool to do this because it’s very interactive and sort of cool to look visualize so it’s bound to catch the eye of most students and facilitate learning about biology more specifically genetics.

Although this isn’t sort of a direct impact on the environment like the two mentioned before I think it’s still important to mention that there is environmental education value in our application in terms of evolution and species genetics. bioMorph mimics a lot of these real evolutionary processes found in nature and the environment so there is certainly a connection between bioMorph and the environment. And now let us finally reveal our application and demo it for you.