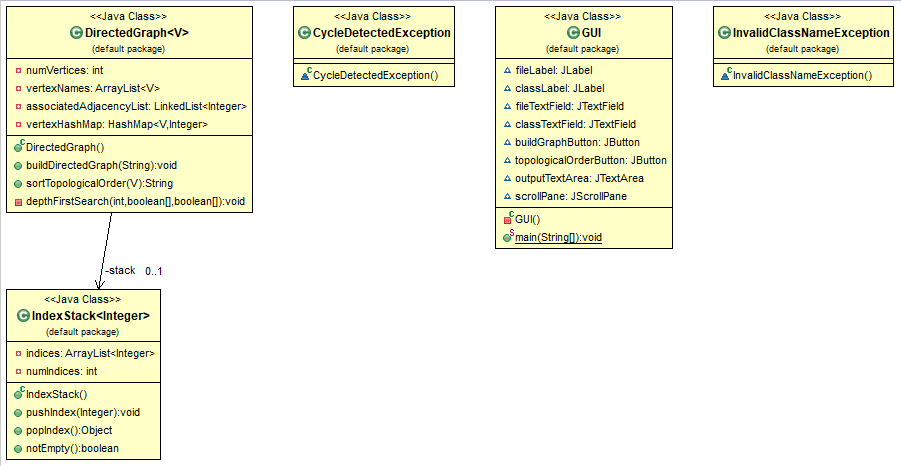
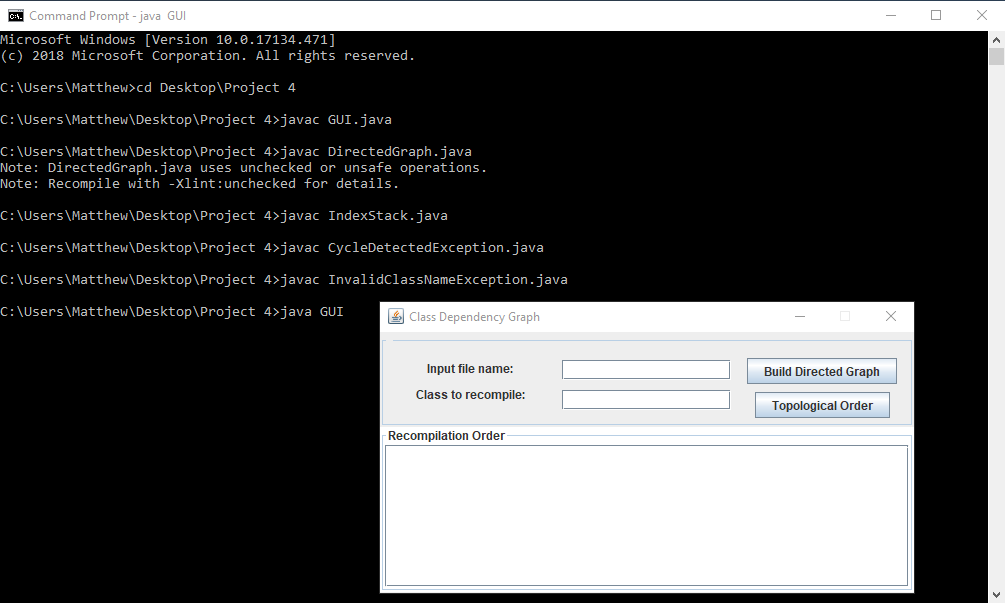
**UML Class Diagram**



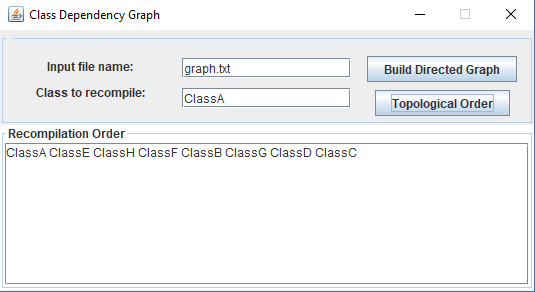
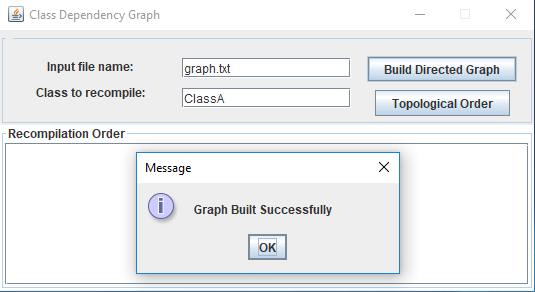
**Test Case Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Expected Output** | **Actual Output** | **Pass?** |
| **Test Case #1: Graph Without Cycles**  **Input file name: graph.txt**  **Class to recompile: ClassA** | **Message**  **“Graph Built Successfully”**  **ClassA ClassE ClassH ClassF ClassB ClassG ClassD ClassC** | **Message**  **“Graph Built Successfully”**  **ClassA ClassE ClassH ClassF ClassB ClassG ClassD ClassC** | **Yes** |
| **Test Case #2: Same File, Different Class**  **Input file name: graph.txt**  **Class to recompile: ClassE** | **ClassE ClassH ClassF ClassB ClassG ClassD** | **ClassE ClassH ClassF ClassB ClassG ClassD** | **Yes** |
| **Test Case #3: No Input File**  **Input file name:**  **Class to recompile:** | **Error**  **“A File Name is Required"** | **Error**  **“A File Name is Required"** | **Yes** |
| **Test Case #4: Invalid File Name**  **Input file name: gr@ph.txt**  **Class to recompile:** | **Error**  **“File Did Not Open”** | **Error**  **“File Did Not Open”** | **Yes** |
| **Test Case #5: Invalid Class Name**  **Input file name: graph.txt**  **Class to recompile: ClassX** | **Error**  **“Invalid Class Name: ClassX”** | **Error**  **“Invalid Class Name: ClassX”** | **Yes** |
| **Test Case #6: Graph With Cycles (I)**  **Input file name: cyclegraph.txt**  **Class to recompile: ClassA** | **Message**  **“Graph Built Successfully”**  **Error**  **“The Graph Contains a Cycle”** | **Message**  **“Graph Built Successfully”**  **Error**  **“The Graph Contains a Cycle”** | **Yes** |
| **Test Case #7: Graph With Cycles (II)**  **Input file name: cyclegraph2.txt**  **Class to recompile: ClassC** | **Message**  **“Graph Built Successfully”**  **Error**  **“The Graph Contains a Cycle”** | **Message**  **“Graph Built Successfully”**  **Error**  **“The Graph Contains a Cycle”** | **Yes** |

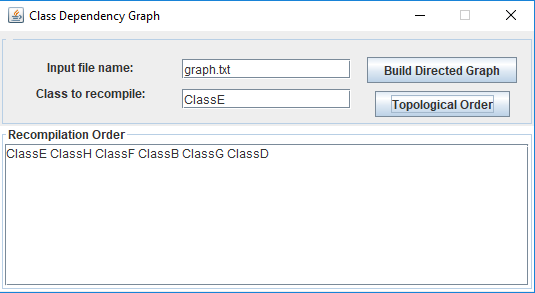
**Screen Capture of Successful Compilation & Execution of Java Code**



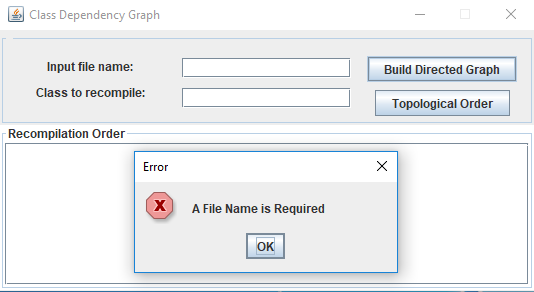
**Screen Capture of Test Case #1**



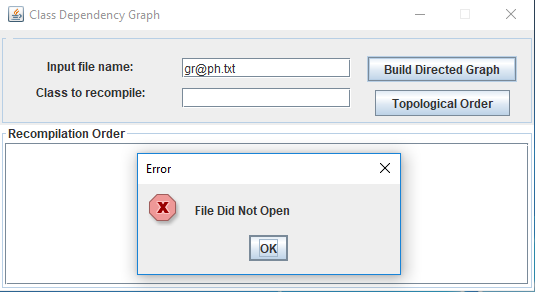
**Screen Capture of Test Case #2**



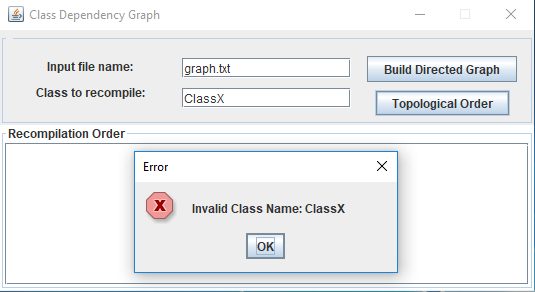
**Screen Capture of Test Case #3**



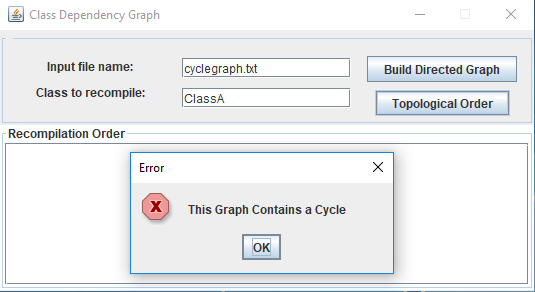
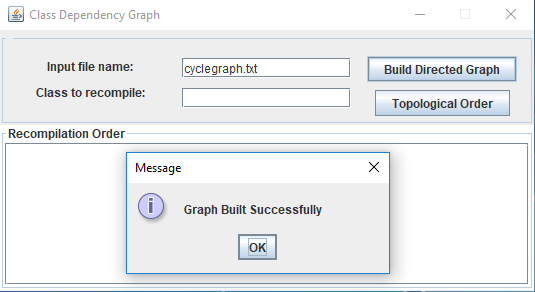
**Screen Capture of Test Case #4**



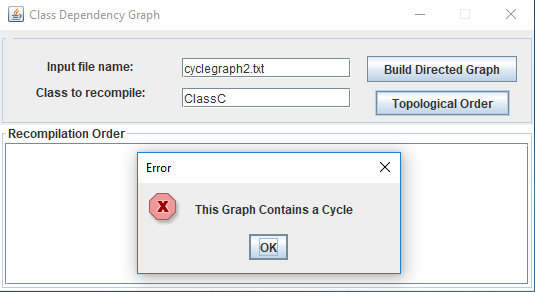
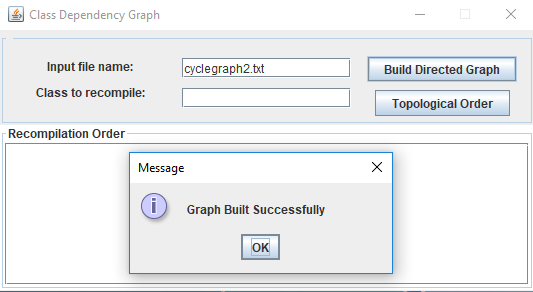
**Screen Capture of Test Case #5**



**Screen Capture of Test Case #6**



**Screen Capture of Test Case #7**

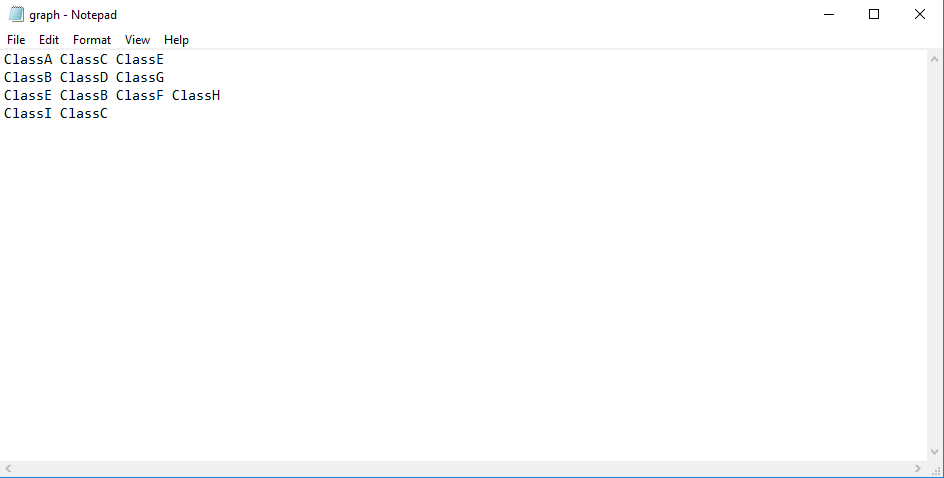


**Lessons Learned**

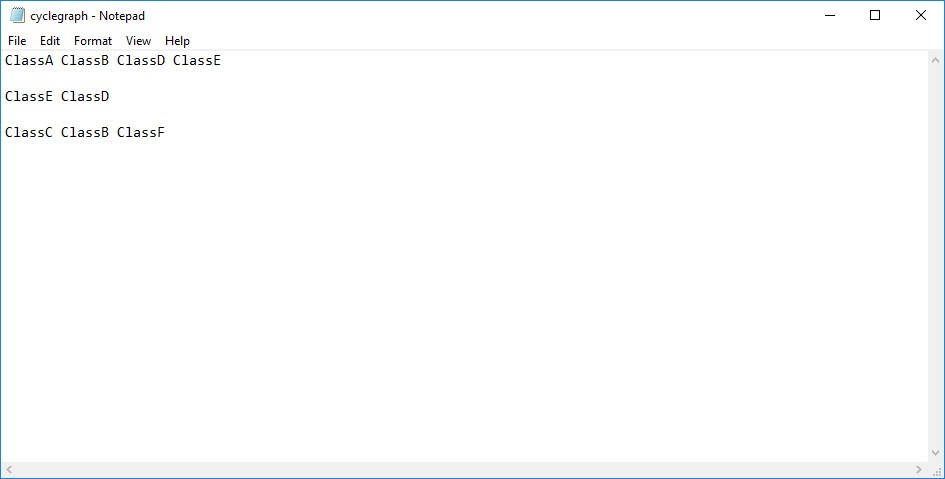
This was a pretty difficult project for me, but I feel that I learned quite a bit from working on it. The GUI interface took me longer than it should have to create it, but in so doing I was able to experiment with various layouts associated with Java Swing. I also got to use generic types again, which is something that I was a bit rusty on. I think the most important thing I learned from this project was how to implement a directed graph in Java.

**Graphs Used**

**graph.txt**



**cyclegraph.txt**



**cyclegraph2.txt**

