**Nim Peer Reviews**

**See below for your peer review assignments. Your name is in the left column and your peer review assignments are in the right column. Each author’s name is a link to download their program. Copy the code into a test.java file in Eclipse and change the Class name to match your test class name.**

|  |  |  |
| --- | --- | --- |
| **Author Name and Link** | **Review 1** | **Review 2** |
| [**Aiden**](https://drive.google.com/file/d/1xZxtEuwiSL9fRe4kQ55h8pYYYui-E6AC/view?usp=sharing) | Dingyi | Emma |
| [**Dingyi**](https://drive.google.com/file/d/1BdTCRgVvXEJ5IUR4PNobBaxTMesxfuM4/view?usp=sharing) | Emma | Gavin |
| [**Emma**](https://drive.google.com/file/d/1_k05_vpFx5pmDm5V6u2625mjFpX7j7o8/view?usp=sharing) | Gavin | Ife |
| **Gavin** | Ife | Ikon |
| [**Ife**](https://drive.google.com/file/d/1zXX5O3inWgsIGKdAfX-2B4tGm7yoyhZS/view?usp=sharing) | Ikon | Kyle |
| [**Ikon**](https://drive.google.com/file/d/1H_3oq9WjIJeZzlKJWAuJFVX2hV1LeeM0/view?usp=sharing) | Kyle | Matthew |
| [**Kyle**](https://drive.google.com/file/d/11ZkaTcjDgNI_l_-MRr5c4UpWRdcad4NS/view?usp=drive_link) | Matthew | Nikhil |
| [**Matthew**](https://drive.google.com/file/d/13U8DOlSG_U4-LMwcrvorXYL6SOmU67aA/view?usp=sharing) | Nikhil | Stephen |
| [**Nikhil**](https://drive.google.com/file/d/16V3mhPhQ0F-eTNoJoFAFK62R_cBdjO4x/view?usp=sharing) | Stephen | Zoey |
| [**Stephen**](https://drive.google.com/file/d/17n5G_FAEC0GuHG6nfABrZVR-VkW5Ubbm/view?usp=sharing) | Zoey | Aiden |
| [**Zoey**](https://drive.google.com/file/d/1E6saqKPIdKNKV5vsKKebdWuU2OAa5Yat/view?usp=sharing) | Aiden | Dingyi |

Make two copies of this document and share your feedback with the author of the program. Submit both copies on Canvas for the Nim Peer Review Assignment.

Author Name:

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Points Available** | **Points Earned** | **Comments** |
| **Creates process to decide and track order of play** | 3 Points | 3 |  |
| **Player and computer remove items from pile (pile count remains correct throughout game)** | 4 Points | 4 |  |
| **Player/Computer can only remove 1, 2 or 3 items** | 3 Point | 3 |  |
| **Player/Computer cannot remove all items from pile** (pile size cannot be negative) | 2 Points | 0 | Protective programming could be bypassed player could enter the same number two times and the pile would be 0 but the program is not ended. |
| **Program ends when pile reaches zero** | 2 Points | 0 | As mentioned above |
| **User Interface/Experience**  (Program instructions are clear and easy to understand. Players receive feedback (pile size) after every turn and after the game ends. Game does not crash on bad user inputs.) | 4 Points | 3 | Some minor grammatical errors, and some instructions are not clear when reentering. |
| **Program Organization/Structure**  (Code is organized and easy to understand. Comments are provided to explain logic that may be unclear.) | 2 Points | 2 | Comments are provided, although some code could be organized using comments and methods it is not challenging to understand |
| Total Points | 20 Points | 16 |  |

* What is one feature of this Nim program that you really liked?

The user gets a choice to pick how goes first, the result is not determined.

* What part of the Nim program seemed confusing or did not work as intended?

Some protective programming failed the second time.

* Make a suggestion to make this program better:

Create a algorithm so that the computer plays like a human, and let the user choose whether they want to go first without seeing the pile, so that it is more undetermined and more interesting.