---Readability Review---

Author of the code review: Kevin Nguyen
Date of the code review: 3/3/22
Sprint number: 3

Name of the .cs file being reviewed:
MouseController.cs

Author of the .cs file being reviewed: Chris Johnson
Number of minutes taken to complete the review: 2

Specific comments on what is readable and what is not:
The Update() function has if-else statements that are well written and easy to read. Functionality of the class is easily understood.

---Code Quality Review---

Author of the code review: Kevin Nguyen
Date of the code review: 3/3/22
Sprint number: 3

Name of the .cs file being reviewed:
MouseController.cs

Author of the .cs file being reviewed: Chris Johnson

Specific comments on code quality:
Effective use of if-else statement conditions to minimize the amount of code. Helps to make the file easier to follow

A hypothetical change to make to the game related to file being reviewed and how the current implementation could or could not easily support that change
There are no changes necessary, and the current implementation can be easily supported in the future.

---Readability Review---

Author of the code review: Kevin Nguyen
Date of the code review: 3/7/22
Sprint number: 3

Name of the .cs file being reviewed:
Door.cs

Author of the .cs file being reviewed: Chris Johnson
Number of minutes taken to complete the review: 2

Specific comments on what is readable and what is not:
The switch-case has good readability and the constructor is straight forward in setting its class fields.

---Code Quality Review---

Author of the code review: Kevin Nguyen
Date of the code review: 3/7/22
Sprint number: 3

Name of the .cs file being reviewed:
Door.cs

Author of the .cs file being reviewed: Chris Johnson

Specific comments on code quality:
All functions in the file serve one purpose and don't depend on too many variables, which is good.

A hypothetical change to make to the game related to file being reviewed and how the current implementation could or could not easily support that change
The constructor takes in a lot of arguments, so it is required to know too much information. This might affect coupling in the future.