CPTR 142: Project #1 Name: _____

Code Review

Before you can receive an "M" on your project, you must participate in a code review with someone who has taken CPTR 242—for example, a C.S. tutor at the Student Development Center. This walk-through style review is a guided-tour of your source code in which you describe how you implemented the various features, explain why you made the choices you did, and solicit constructive input which might help improve your final product.

Please discuss with your reviewer how your work lines up with the rubric below (The required concepts are on the back of this page). Submit the completed form to your instructor.

Reviewer Name _____

Reviewer Signature _____

Date of Review _____

When did reviewer take Data Structures? _____

	Algorithm	User Interaction	Use of Required Concepts	Coding Style
E	Handles a simulated game for N players	Excellently formatted output and handles input errors well	All required concepts included	Code is well commented and readable
M	Handles a simulated game for a set number of players	Good formatting and handles most input errors	All required concepts included	Code is sparsely commented or over-commented but mostly readable
R	An attempt to implement the game was made, but was not successful	Basic user interaction with little or no input validation	Some required concepts were used, but not all	Comments are sparse or non-existent and the code is very hard to follow
N	No attempt at implementing the game is evident	No user interaction or input validation	Only one or two required concepts were included	Comments are lacking and/or code is unreadable

Summary of Feedback _____

Solution Specification

Your solution should strive to meet the standards specified below as they form the basis on which it will be graded.

1. Your program must prompt the user for input in a meaningful way and check for possible error conditions.
2. Have the user enter a seed for the random number generator so that your runs are repeatable.
3. Since this is a project, you have a lot of freedom in how you choose to accomplish the task. However, you are expected to make use of the following concepts somewhere in your program.
    - Appropriately named variables
    - Organize your code into classes and functions
    - Split your code into multiple files
    - Create unit test to validate your code's functionality