The George Washington University

CSCI 3410 Systems Programming

Homework 3 - Let's Play a Game!

Due Date: Feb 27, 2017 at 6:00 PM

You may work in teams of *up to* **three (3)** members from your lab section

Background Information

This assignment extends what we started in Lab 5. In your groups, you will write programs for users to compete in a game of your choice over I^2C. The game you choose must satisfy the following requirements:

- It must be a *2 player turn-based game*.
- During each turn, *both* players must input a meaningful decision. Thus, a game like "war" (where each player simply draws from a deck of cards) does not qualify.
- The game needs some notion of scoring and/or a victory condition

For example, here are a few valid games you may choose, as well as their complexity points (factored into your grade). If you like, you may select a different game. You can email the TAs before the deadline to find out how many complexity points it's worth.

Table 1: Pre-Approved Game Complexities

Game	Complexity (max 15)
Go fish	0
Blackjack	5
Tic-tac-toe	10
Poker	15
Battleship	15

The following Arduino tutorials and wiring diagram provide a good example on basic I^2C communication:

- Master Writer
- Master Reader

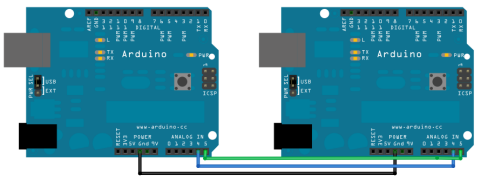

Figure 1: Two Arduino I^2C Bus

Specification

You must build an I^2C bus with as many Arduinos as you have group members. If you are a group of three, the third Arduino must act as a "judge" and control the game flow. You must decide how to organize communication among the Arduinos on the I^2C bus and how to devise a state machine for your chosen game. Make use of the Serial monitor for input and output to your programs. You will also need to implement the scoring mechanism and victory conditions for your game. After each turn, each player should output the current overall "game state", which may include the scores of each player.

Deliverables and Grading

- Push your code to GitHub (a link will be posted on Blackboard) before the deadline. The following things should be in the repository:
  - Two Arduino sketches, one for each player.
  - If your group has three members, one additional Arduino sketch for the "judge"
  - A document (in `pdf` format) explaining which game you chose, how you implemented your solution, details of your I^2C communication protocol, and a flowchart for the game's state machine
- You will be giving a demo of your circuits before class on the day of the deadline. Be sure to bring everything you need to show it off.
- Be sure to write clear and concise commit messages outlining what has been done.
- Write clean and simple code, using comments to explain what is not intuitive. If the grader cannot understand your code, you will lose credit on the assignment.
- Be sure your code compiles! If your sketches do not compile, you will receive **no credit**. It is better to submit a working sketch that only does a subset of the requirements than a broken one that attempts to do them all.

Table 2: Grading Rubric

Category	Percentage
Demo	60%
Write-up	20%
Code Quality	20%
Game Complexity	15%