Test 1 - Programming

COMP 441

Fall 2020

60 points

Please read the instructions below completely:

* **Your project must build and run without crashing.** If your project does not build or run, you may receive zero credit.
* When you are finished, upload a zip of the code to my.gcc, **after deleting the “Library” folder** to make it smaller. You *must* include code, and not just an executable.

A word of clarity: For any game I suggest, there’s going to be a “How to make game X in Unity” tutorial. You are welcome to use tutorials on Unity features; you are *not* welcome to swipe someone else’s solution to this exact problem. Doing so is an academic integrity violation.

**Quick Summary**

You are going to make a simple version of a game of digital pool.

1) You must include:

* A cue script, to handle the cue ball
* A pocket script, to handle balls landing in the pockets
* A MainSceneManager, to handle spawning and score.

All other functions may be distributed as you see fit. (5 pts for these objects handling these functions)

2) Surround the playable area in a box bumpers on all sides that block movement. (5 pts)

3) Within the playable area, create a white “cue ball” and 3-4 billiard balls of varying colors, as well as a black “pocket” in at least once corner. A score should display in another corner. (Be sure the score will still display for me, even if our resolutions are different.) (5 pts)

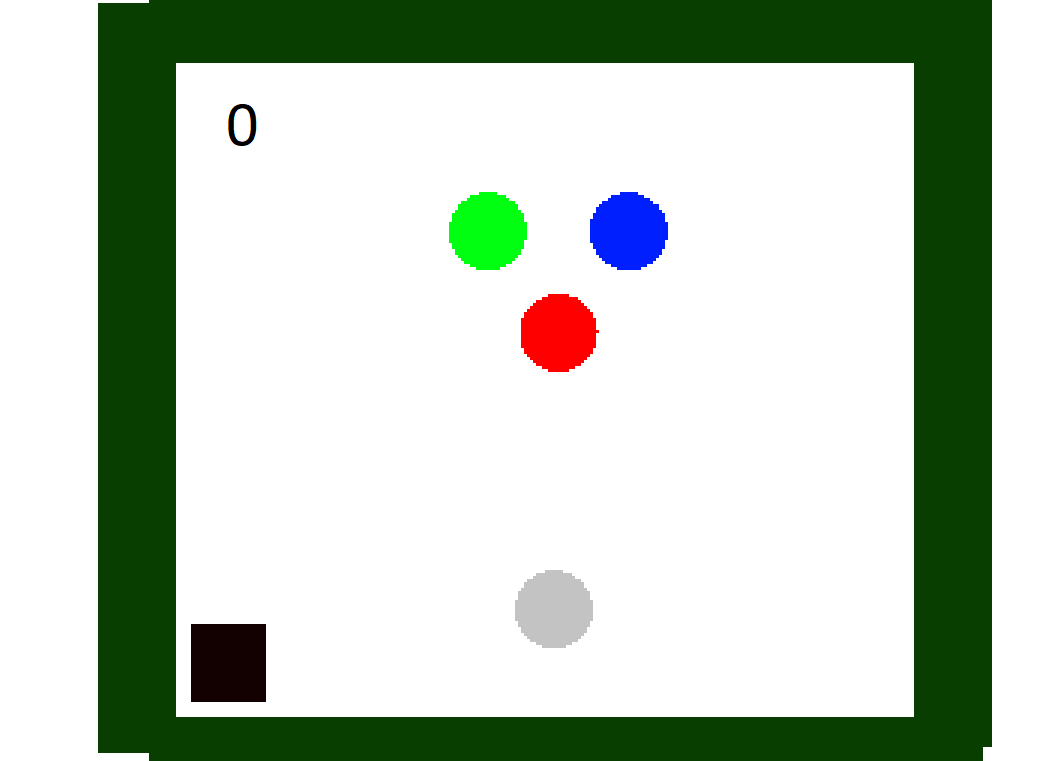
4) Allow the player to move the cue ball left or right on the screen by holding the respective arrow keys. Do not allow him to control movement up and down. (You can choose what happens if the player hits the left-right keys while the ball is already in motion; there are several valid approaches.) (10 pts)

5) When the player holds down the space bar, he “charges” the cue ball. When he lets go of the space bar, the ball should fly upward; the longer the ball charged, the faster it should go, in some reasonable range. (10 pts)

6) The cue ball and the other balls should bounce off each other satisfyingly, and should not be subject to gravity. (5 pts)

7) If a ball (other than the cue) ricochets into the pocket, destroy that ball and award the player one point. If the cue crashes into the pocket, destroy the cue and subtract a point. The main scene manager should handle updating the score. (10 pts)

8) If the cue ball is destroyed in this way, the main scene manager should spawn a new one at some convenient location. (10 pts)



*An example of what your game might look like. The cue ball is in the lower center; the pocket is in the lower left. The balls are in the upper center. The score is currently 0.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement** | **Comments** | **Points Earned** | **Points Possible** |
| Cue, pocket, and main scene manager scripts are all in place. |  | 5 | 5 |
| Surrounding box is in place. |  | 5 | 5 |
| Balls, pocket, and score are in place. |  | 5 | 5 |
| Player can move cue left and right. |  | 10 | 10 |
| Player can “charge” cue with spacebar. |  | 10 | 10 |
| Ball physics is reasonable. |  | 5 | 5 |
| Balls disappear on impacting pocket, scoring points. |  | 5 | 5 |
| Cue disappears on impacting pocket, losing points. |  | 10 | 10 |
| Cue respawns if destroyed. |  | 5 | 5 |
| **TOTAL** |  | **60** | **60** |