

Homework #4: Russian Blocks

Due: Tuesday, February 24 @ 12:30 PM

Submission:

Please turn in all files on Canvas before the deadline. You should compress your submission into a single file, do not submit a large number of individual files. If you know you are going to miss a deadline, contact the TA **before** the deadline. If you do not do this, skip days will automatically be deducted. Canvas has been known to be quirky, so it is not advised to wait until 5 minutes before it is due to make your submission.

Please include a text file called “README” at the top level of your main project directory. Include the following:

- Your name
- Your email address
- How long this project took you to complete
- Any comments or notes for the grader
- The type of iphone I should use to run your project

Overview:

This is not a group assignment. It is acceptable to consult with other class members, **but your code must be your own.**

Using a framework provided to you, you will construct a simple Tetris-like game.

For those who have never played tetris, definitely try it out. Below is a link to a free online tetris game. Use the “up” arrow key to rotate the blocks.

<https://www.freetetris.org/game.php>

Specifications:

Definitions:

Cell: this term refers to one 1x1 block item on the grid. This does not refer to empty spaces on the grid.

Tetronimo: this term refers to a collection of four blocks. There are seven varieties of tetronimos.

Part 1: Model

Provided for you are several classes that should greatly assist you in constructing your game.

Feel free to use these classes in part, in whole, or not at all. You may also modify them as you feel necessary.

Although some documentation is provided below, it is very likely that you will need to peek into the source code to understand what is going on.

Coordinate

Basic representation of a position of a block on the screen. Internally represented by two integer values. Create a new coordinate object with **[Coordinate X:(int) Y(int)]**

Cell

The object that represents a single block on the screen.

Grid

A two dimensional array of cells. Provides basic functions such as adding, removing, and testing for cells. It is important to note that grid position (0,0) is in the lower left hand corner.

Your grid in this project should be exactly 10 wide and 22 tall. The top 2 rows should not be rendered in your view. (This is where you will add new blocks.)

Tetronimo

A grouping of cells. Provides basic functions that allow it to be inserted into a grid and to be moved and rotated. The Tetronimo and the grid into which it is inserted both have pointers to the cells. If the Tetronimo is deleted (i.e. there are no more strong pointers to it) then the cells will remain in the grid.

See <http://tetris.wikia.com/wiki/Tetromino>

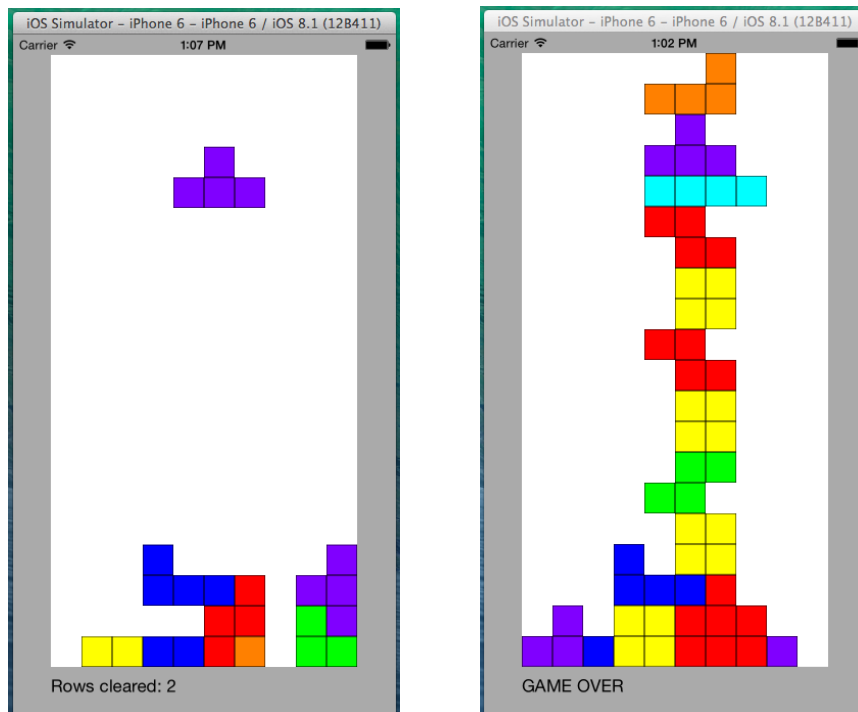
TetronimoBuilder

Use this class to help construct Tetronimos in the 7 basic shapes and colors.

If you find any bugs in the provided code, make a posting to piazza. The first student to report a particular bug will receive 1 point extra credit on this assignment.

Part 2: View

You are implementing a basic game of tetris. Your UI should look something like the images provided below.



The main screen should be 10 blocks wide and 20 blocks tall. This means that the width of this particular view should be **exactly** twice as tall as it is wide.

Don't forget to add some sort of border to cells as shown above.

Swiping left or right with one finger should move the tetronimo one position to the left or right.

Swiping up with one finger should rotate the tetronimo right (clockwise).

Swiping down with one finger should cause the tetronimo to “zoom” down to the bottom. See <http://tetris.wikia.com/wiki/Drop>

The pictures above show the bare minimum required for this project. There are several additional features that you may choose to implement for extra credit. (Described below)

Part 3: Control

Your program should implement the basic behaviors of tetris. For more information, either play the game or read about the rules at http://tetris.wikia.com/wiki/Tetris_Wiki

As long as the tetronimo is still falling, the player should be able to move or rotate it any time.

Every 0.5 seconds or so you should move the active tetronimo down by one position. If the tetronimo cannot move down due to “gravity” then it becomes locked and can no longer be shifted by the player.

Since multithreading has not yet been discussed, you do not need to concern yourself about potential race conditions between gravity and user actions.

Any time there is a row completely filled by cells, that row should be removed and the score increased by one. All cells above that position should be moved down by one position.

When the blocks “overflow” over the top of the screen, the game is over.

Extra Credit:

The following things may be implemented for extra credit. If you implement any of these things please make a note of it in your readme.

- Reset button
- Tetris music
- Sound effects when blocks are shifted and rotated
- Ghost piece http://tetris.wikia.com/wiki/Ghost_piece
- Piece preview http://tetris.wikia.com/wiki/Piece_preview
- Levels with increased difficulty
- A true score as well as a line count <http://tetris.wikia.com/wiki/Scoring>
- A more polished user interface (basically, does your UI look significantly better than the example provided)

Notes:

You may use any code presented in class. Please type the code yourself (as opposed to copy-paste) as it is a good learning experience.

If you have time, watch this video: <https://www.youtube.com/watch?v=hWTFG3J1CP8>

It may be worthwhile to investigate **[NSTimer scheduledTimerWithTimeInterval:...]**