Tetris Implementation Classes and whatnot:

Suggestions:

* For integration and version control we should use git.
* Push to git very often.
* Abide by the class names and methods that we decide upon
* Gravity is generally given as a fraction IG/256, where IG scales with level. This number correlates with rows per frame. i.de. When IG = 256, the Tetromino is moving downwards at a rate of one row every frame. (This formula and numbers come from <http://tetrisconcept.net/wiki/Tetris_The_Grand_Master> I don’t believe it corresponds to the original tetris game, but presumably it corresponds to one that was similar.
* There is a whole wiki dedicated to different versions of tetris and various mechanics therein. This will be particularly useful for defining rotation algorithms and Tetromino superclass. (tetris.wikia.com).
* We have some liberty in our implementation, but a significant amount of legwork has already been done on the wiki page and other pages that definitely lead to a fairly playable game
* Lots of Tetris guidelines can be found at http://tetris.wikia.com/wiki/Tetris\_Guideline

Classes:

1. Tetrimino superclass (All four block pieces are tetriminos):
   * Tetriminos spawn in a particular orientation and color depending upon their shape
   * Boolean ActiveState
   * Block[]

* Color
* Drop (same for all tetriminos)
  + Key Listeners/mechanics for hard drop/soft drop
* Move
* Rotate

1. I\_Piece extends tetrimino:

* Color = Red

1. J\_Piece extends tetrimino:

* Color = Magenta

1. L\_Piece extends Tetrimino:

* Color = Yellow

1. O\_Piece extends Tetrimino:
   * Will not really need a rotate() function, as it’s appearance and function are identical regardless of position

* Color = Cyan

1. S\_Piece extends Tetrimino:

* Color = Blue

1. Z\_Piece extends Tetrimino:

* Color = Green

1. T\_Piece extends Tetrimino:

* Color = Light\_Gray

1. Block class:

* Color
* Width
* Height
* Location

1. GameWindow extends JFrame:
   * Pirate sweet tetris music perhaps
   * GamePanel
   * ControlPanel
   * NextTilePanel
   * StatsPanel
   * Key Listeners to drop, move and rotate the active Tetrimino
2. GamePanel extends Panel:
   * Displays the Tetriminos
   * Keeps track of gravity as a function of level
   * Keeps track of rows
   * Ghost Piece? (Seems like it would be pretty easy to implement, not particularly necessary)
3. ControlPanel extends Panel:
   * JButton pauseButton – Stop time, stop movement, hide pieces
   * JButton newGameButton – Pauses and Spawns a modal asking if you’re sure
4. NextTilePanel extends JPanel
   * Displays the next Tetrimino on the Tetrimino Queue
   * There is also a mechanic to reserve a Tetrimno for later use.
5. StatsPanel

* JLabel to keep track of level
* JLabel to keep track of score
* JLabel to keep track of rows removed
* JLabel to keep track of play time