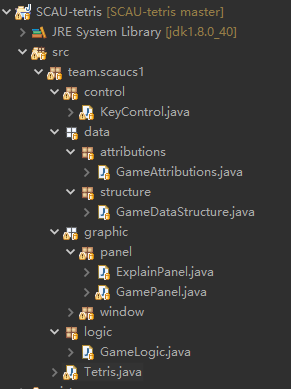
**SCAU CS1班 小组数据结构课设项目**

**俄罗斯方块**

项目地址(按住Ctrl访问链接)：[Galaxy-wkx/SCAU-tetris (github.com)](https://github.com/Galaxy-wkx/SCAU-tetris)

推荐使用github desktop软件，方便下载和上传代码，具体可以百度学习一下（功能相当于一个在线云文档，这样我们做的代码改动可以同步更新）。

项目结构：



一、data层：

1.attributions属性类

里面定义游戏的常数属性信息，比如窗口宽度高度、游戏区域的行列数、线程的休眠时间、消除一行后奖励得分等等。

2. structure数据结构类

里面存储游戏用到的数据结构即一些方法。主要是两大结构。

第一个是二维数组矩阵matrix[][] 模拟游戏区域，1代表有方块，0代表空。

第二个是allRect[] 数组，里面通过4位16进制数来存放一个图形信息。由于俄罗斯方块规则决定了生成的图形大小不会超过4X4，因此一个4位16进制数足以表示一个图形。例如其中的一个rect = 0x00cc，将其转为二进制串为，并且规定最高位作为首行，第二位作为第二行以此类推就得到一个：

这样的图形。

另外还提供一些方法，比如通过随机数产生索引，生成一个随机图形。

以及通过一个4位16进制数(对应一个图形)获取图形有方块位置的坐标，具体实现过程大概是这样的：比如还是这个rect = 0x00cc，如果以左上角为x=0, y=0，那我们知道，有方块的地方的坐标是(2,0),(2,1),(3,0),(3,1)四个点，用程序逻辑实现，我们需要一个singleRect = 0x8000，由前面的知识，这个singleRect的图形是这样子的，如果将singleRect = 0x8000与rect = 0x00cc进行按位与运算，得到的结果会是0，因为singleRect只有最左上角有一个方块，但rect这个地方没有方块，这也说明了对于rect图形，(0,0)这个坐标上没有方块。之后把singleRect右移一位(>>操作符)，就变成了这个样子，继续按位与运算，结果是0，结论是(0,1)这个坐标上没有方块，这样不断右移16次，哪一次按位与结果不为0，说明那次对应坐标有方块，这样就找出rect图形所有方块的坐标了。

二、graphic层：

1.panel层，面板，游戏区域的面板，说明信息的面板。

2.window层，主要通过JFrame创建窗口，调用到各个面板。

三、logic层：

主要实现游戏的逻辑，大概是这样：生成一个新图形，先判断能否下落，底下没方块挡住则可以下落，那么进行下落一格的操作。

如果下面有方块说明到底了，那么不能继续下落，图形锁定，它的信息写入matrix矩阵中固定住，同时这时候检查有没有行满了，如果满了就消除这行，奖励得分，并且加快掉落速度（提高难度）。最后在检查一下是不是已经堆到最高了，堆到最高游戏结束，否则游戏继续，重新生成新图形，循环……

四、control层

主要是监听键盘，根据键盘指令进行图形移动和变形操作，要注意在变形和移动之前，需要判断是否会出界。

最后就是启动器主类Tetris了，用来启动游戏的。

**优化思路：三个方面一人做一个**

**1.界面层：让游戏变得更好看，考虑增加贴图，以及让方块更多彩。**

**2.逻辑层：增加游戏难度设置，游戏玩法设置（眩晕、冻结、反向、加速、减速……）**

**3.控制层：配合逻辑层实现相应的新功能。**