

玩法和普通的2048一样。

计分法则：目前的数字总和

分数位于最上方居中

一旦游戏失败，下方信息框显示game over

重要参数的意义：

contentPane:图形界面

grid:JLabel的list，共16个，用来画数字格

gridint:grid对应的中间数字的int表示，方便操作

grids:一个JPanel，由grid中16个小格子组成

isfilled:boolean型数组，用于判断各个小格子是否被占据

label:用于显示分数的JLabel，位于north位

labeldown:用于显示游戏状态，be happy表示游戏进行中，game over表示游戏失败，位于south位

lose:用于标记游戏状态，true表示游戏结束，false表示未结束

各个函数的用处：

canmove:判断是否能够继续移动，同时会改变lose的真假，若不能移动了就为true，否则为false

canmovedown:判断是否能够下移

canmoveleft:判断是否能够左移

canmoveright:判断是否能够右移

canmoveup:判断是否能够上移

drawgrid:画中间格子界面以及显示分数

getscore:计算当前数字总和作为分数

movedown:下移操作，在canmovedown时先下移消空，然后找可以下移合并的块进行合并，每一次合并后再做一次下移消空，移动完成后利用producegrid产生新数字，然后canmove更新lose值，最后drawgrid画界面

moveleft:类似下移操作

moveright:类似下移操作

moveup:类似下移操作

numofblank:空格子的总数

producegrid:随机在空格子产生一个新数字，值为随机的2或4

removedownblank:下移消空

removeleftblank:左移消空

removerightblank:右移消空

removeupblank:上移消空

构造函数game2048:

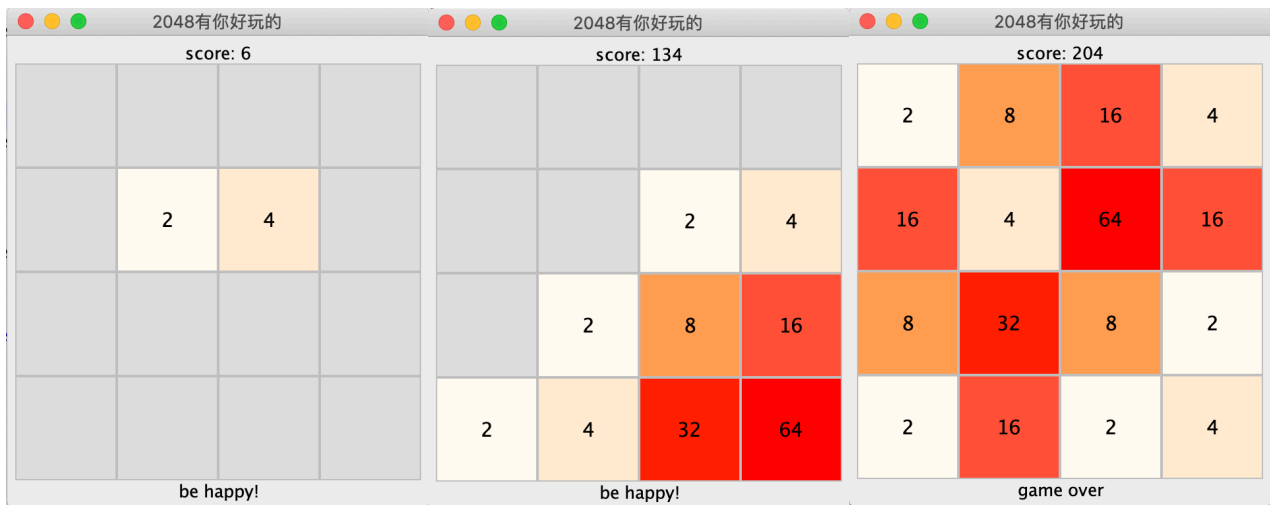
先画基本的界面，标题为“2048有你好玩的”，设置关闭窗口退出，设置边界大小；

contentPane是block布局，north是显示分数的label，center是格子界面grids，south是显示游戏状态的labeldown；

grids是grid布局，有4\*4个小格子；这些小格子中设置字体居中等属性，小格子被设置成不透明，大小80\*80；

然后产生一个2在随机的一个格子里，再利用producegrid随机在空格子产生一个新数字，值为随机的2或4，这就是初始状态，调用drawgrid画出来；

然后是一个键盘时间监听，判断按下的键，上下左右键分别进行相应的move操作，然后判断是否lose，lose就labeldown显示game over。



初始状态示例

游戏中示例

结束示例