**PROJECT2设计文档（16302010061曹佳颖）**

一、面向对象设计

**1、GameController类**

(1)作用：调用别的类中方法来控制游戏

(2)父类子类接口：无

(3)构造函数，可见域和可见方法：

构造函数：无参构造函数（正常开始游戏）和GameController(int number, View view)（读档开始游戏，其中第一个参数表示是第几个存档）

可见域：无

可见方法：public void play()：游戏主体，完成记录每一步的棋盘、动物移动、切换玩家

public static void undo()：悔棋，调用History类中的悔棋方法。

public static void save(int number, View view)、public static void load(int number)：存档读档，调用History类中相应方法。

public void setAnimal(int x, int y)、public void setDirection(int x, int y)：设置每次要移动的动物和方向。

public boolean attention() ：判断是否有危险。

public boolean ifEnd()：判断是否结束。

(4)私有域：Board gameBoard：当前棋盘。

History historyBoard：历史棋盘。

Animal animalChosen，int[] direction：要移动的动物和方向。

(5)私有方法：无。。

(6)其他：无。

**2、Board类**

(1)作用：通过读取文件生成开始的棋盘，并记录动物位置的信息。

(2)父类子类接口：无。

(3)构造函数，可见域和可见方法：

构造函数：无参构造函数（正常开始游戏）和Board(int save,View view)（读档开始游戏，其中第一个参数表示是第几个存档）Board(int[][] map)（前两个构造方法会调用这个构造方法，传入的参数通过读取文件获得的地图）

可见域：public static int[][] landForm，public static Animal[][] animalBoard分别记录地形和动物。

可见方法：无。

(4)私有域：无

(5)私有方法：setAnimalBoard(int[][] map)：根据读取的数字型地图生成相应的动物地图。

readMap()、int[][] readAnimalMap()、int[][] readSaveMap(int number, View view)：分别读取地形地图，动物地图，存档的动物地图。最后一个的参数view是为了初始化view中的玩家和计时中的时间。

(6)其他：无。

**3、History类**

(1)作用：记录每一步的棋盘，提供悔棋、存档、清空的方法。

(2)父类子类接口：无。

(3)构造函数，可见域和可见方法：

构造函数：默认的。

可见域：无。

可见方法：addBoards()：每走一步把当前的动物棋盘加进去。

save(int number, View view)：存档。分别存最后一次的棋盘（包括当前玩家和时间）和之前所有的棋盘。

undo()：悔棋，顺便把悔过了的棋盘清除。

loadHistory(int number)：读档的时候把之前的每次走过的棋盘都重新加到animalBoardRecord里（为了实现读档后悔棋）。

clear()：重开游戏时清空棋盘。

(4)私有域：ArrayList<Animal[][]> animalBoardRecord：记录每一次的动物棋盘。

(5)私有方法：无。

(6)其他：无。

**4、Animal类**

(1)作用：

(2)父类子类接口：

子类：SpecialAnimal中的Lion,Tiger,Mouse类，重写了move方法，Lion和Tiger中重写jumpRiverDirection(int[] direction)方法改变遇河时该方向的下一位置。

(3)构造函数，可见域和可见方法

构造函数Animal(int value, int x, int y)：生成一直值为value,坐标为（x，y）的动物。

可见域：无。

可见方法：getters和setters

boolean move(int[] direction)：判断该方向上动物能不能动。

go(int[] direction)：向该方向移动。

(4)私有域：

Int camp：-1表示左方，1表示右方，0表示空白。

int value：动物的值，空地的值为0.

x,y ：坐标。

(5)私有方法：无。

(6)其他：无。

**5、Judge类**

(1)作用：全是静态方法，做各种判断。

(2)父类子类接口：无

(3)构造函数，可见域和可见方法：

构造函数：默认。

可见域：无。

可见方法：判断是否出界、是否是河，是否是家，是否有陷阱，是否是敌人，是否能吃，是否能跳河，是否家被占领了，是否死光了，是否被困死了，是否有危险……

(4)私有域：无。

(5)私有方法：无。

(6)其他：无。

二、图形化界面

Start：开始时的界面，四个button（开始，载入，规则，退出），和背景ImageView

View：游戏界面。组件：Pane,GridPane,BorderPane,Hbox,timeline,ImageView,Label……。

实现方式：

功能按钮：分别调用GameController里各种该方法。

动物的移动：首先animalPane（7\*9）里有三层Label,一层蓝的图，一层红的图，一层动物的图（默认蓝红两层不可见，三层默认都不可点击）。每次只让可以动的动物下面的那层红色setMouseTransparent(flase)，鼠标移动到那上面会显示（setVisible(true)）。点一次动物GameController里的animalChosen选成该位置的动物，之后判断周围可动位置，把相应的蓝色setVisible(true), setMouseTransparent(flase)。此时再点击该动物则取消选择，一切恢复最初。若点击了蓝色方块，则传入方向，并移动。之后判断是否胜利、显示棋盘。

三、拓展功能

用来评分的三个功能：

1、智能化提示：鼠标移动到可以操作的动物上会变红，点击后选中的动物会变红，可以移动的位置会变蓝（并且只有变色的地方才能点）；当轮到自己走且自己有危险时会有提示。

2、存档读档：提供了三个存档，当前玩家、当前总时间可以记录，单方玩家的计时会在读档后重新变为15秒，读档后可以悔棋。

3、计时：记录游戏总时长，限制单方操作时间15秒，过后自动切换玩家。

做着玩的希望能加分的功能：

\*4、半成品都不算的愚蠢AI：前面几步走套路，后面自己的动物能吃别人会吃，自己的动物有危险会逃，都没有先判断能否攻击（即下一步能吃）且走的那个位置不会有危险，都没有随机移动一步（且保证不会送死）。（以及测试太麻烦可能有未知bug，比如玩家不走切换到电脑走它不会走……动物乱跳之类的？）反正Start里没提供访问AI的按钮，测试的话还是可以直接运行AIView测试的hh

\*5、并不好看的界面：P了很多图也算做过这一项了吧，但是并不精致而且也没有和代码结合所以算了…顺便做了沙漏的GIF，但是这样暂停的时候虽然时间停下了可是沙漏还会动…

\*6、被注释掉的音效：听说音效的地址很神奇，所以试了试添加音效，但是想到这个功能要是做了应该要添加什么停掉音乐的按钮啊之类的，原来的界面没地方了而且怕弄乱，所以还是不要了 = =||

四、问题及策略

1、往GridPane里加动物，中间应该空的地方会自动缩进……然后就往所有空的地方加了一个透明图充数，结果new的时候只new了一个，却加了好多次，就报错。这里报错还都是nosuchmethod…查了好久。后面因为要做智能提示，干脆弄了63个红的蓝的半透明格子往里填。

2、动物的移动。开始做界面的时候完全不知道怎么通过界面去调用逻辑里的方法，也不知道怎么让动物动。最初的构想是每次界面只要把animalBoard记录的信息展示就好了，可是要做鼠标点击就不知道怎么弄。结果后来不知道为什么写着写着就写出来了（就是解决策略我给忘了）

3、问题：timeline每一步都会越来越快。

策略：似乎是因为每一次都在new一个新的timeline，会叠加？最后每次走完后都把当前的timeline给停了。

4、问题：存档读档问题，开始做的是根据列数位置来判断动物的阵营，但存档后位置乱了。

策略：还是用左方字母右方数字来写地图文件，存档的时候如果是左方就把读到的动物的value弄成char然后加96变成字符，否则就直接记录value的值；读档的时候读出来的值大于96的就是左方。

(似乎，没什么大问题了诶……一些奇奇怪怪的小问题忘了……)