

GOCHESS 项目设计文档

181830044 董宸郅

2020.11

概述

- goChess 提供了3种棋局模式，分别是翻转棋、五子棋和自制的“跳棋”。
- 自制“跳棋”灵感来源于五子棋，规则是3个棋子间隔相连即算成功。
- 在游戏中，玩家可随时退出并选择是否保存进度，之后也可选择开始新游戏或从已有进度中继续。
- 棋局的保存通过对应文件来实现，文件名为开始新游戏的时间戳。
- 对于不合法的输入（如菜单选项、棋子位置），予以相应反馈。

模块划分

- 1. Reversi (翻转棋)
 - 翻转棋的游戏逻辑
 - 游戏记录的读取和存储
- 2. FiaR (五子棋)
 - 五子棋的游戏逻辑
 - 游戏记录的读取和存储
- 3. Jumpy (自制“跳棋”)
 - 五子棋的游戏逻辑
 - 游戏记录的读取和存储
- 4. Admin
 - 游戏的各级菜单选项
 - 整体游戏记录的读取和存储

- FYI: 以下仅展示各类的**核心**成员，一些辅助变量、函数不再冗述...

REVERSI

- `Color board[9][9];` //8*8的棋盘
- `string timestamp;` //开始时间
- `int blackCnt, whiteCnt;` //黑子、白子的计数
- `map<pair<int, int>, vector<Position>> attached;` //可落子位置->对应的本方棋子位置
- `bool findAvl(int x, int y);` //查找当前位置是否存在对应的可落子位置
- `void refresh();` //刷新屏幕
- `void layout();` //绘制棋盘
- `void leave(int mode);` //离开游戏，可选保存记录至 docs/Reversi/timestamp.txt
- `Reversi();` <= 两个重载的初始化函数分别对应从头开始和读取记录
- `Reversi(string ts);`
- `int startGame();` // 返回值表明此次的退出状态，正常结束/中途突出

FIAR

- `Color board[9][9]; //8*8的棋盘`
- `string timestamp; //开始时间`
- `int sum; //棋子总数`
- `bool complete(int x, int y); //判断此位置是否成功连线`
- `void refresh(); //刷新屏幕`
- `void layout(); //绘制棋盘`
- `void leave(int mode); //退出，可选保存记录至 docs/FiaR/timestamp.txt`
- `int startGame();`
- `FIAR();`
- `FIAR(string ts);`

JUMPY

- `Color board[9][9]; //8*8的棋盘`
- `string timestamp; //开始时间`
- `int sum; //棋子总数`
- `bool complete(int x, int y); //判断此位置是否成功连线`
- `void refresh(); //刷新屏幕`
- `void layout(); //绘制棋盘`
- `void leave(int mode); //退出，可选保存记录至 docs/jumpy/timestamp.txt`
- `int startGame();`
- `Jumpy();`
- `Jumpy(string ts);`

ADMIN

- //存储三种模式各自的存档集合
- `vector<string> reversi;`
- `vector<string> fiar;`
- `vector<string> jumpy;`
- `void subMenu(int mode);` //次一级菜单
- `void recordMenu(int mode);` //游戏记录菜单
- //保存记录，分别将属于三种模式的记录名称（时间戳）保存至对应的文件中
- `void storeRecords();`
- `void mainMenu();` //主菜单

请选择您的操作序号：

1. 翻转棋
2. 五子棋
3. 自制“跳棋”
4. 退出游戏

请选择您的操作序号：

1. 新游戏
2. 继续游戏
3. 返回主菜单

请选择您的游戏记录：

1. Mon Nov 9 18-04-55 2020
2. Mon Nov 9 16-11-58 2020
3. Sun Nov 8 18-29-46 2020

	1	2	3	4	5	6	7	8
1	---	---	---	#	#	#	#	#
2	---	---	---	---	*	*	#	@
3	---	---	---	---	*	*	*	---
4	---	---	---	*	*	*	---	---
5	---	---	@	#	*	---	---	---
6	---	---	@	@	---	---	---	---
7	---	---	---	---	---	---	---	---
8	---	---	---	---	---	---	---	---

提示：*->黑子 #->白子 @->当前合法棋步

输入坐标(0,0)可直接退出，或(0,1)以存档...

本轮由黑方出子！

翻转棋模式下，当前可落子处会以@标识

请输入放置棋子的坐标（以空格分开）：

THE END...