

## 程序设计报告

本项目应用 easyx 图形库语言，在 Visual Studio 2022（以下简称 vs）中开发。为了尽量减少不兼容，我将所需的头文件包含在此项目中。可支持编译器有：VC6/ VC2008/ VC2010/ VC2012/ VC2013/ MinGW3.4.5/ MinGW4.7.1/ MinGW4.8.1。

### （一）项目实施及部分事项：

为了避免头文件重复包含，自己只写一个头文件 image.h，与 cpp 文件中的函数分开，主要功能是包含绘制棋盘的一些元素、本项目中常用的如 button 函数、色子动画 die 函数等，在 main.cpp 中调用。

交互过程中部分执行函数间插入 Sleep 函数，主要便于玩家反应，但是玩家依旧不能分神超过 2 秒，一般间隔 2.5 秒。

用**结构体**包含了棋子和格子的一些性质，格子性质有坐标、颜色、颜色数量，棋子有所在格子、状态。

输出日志一定程度上会指导玩家，但主要是提示，内容比较简略，因此玩家需要仔细阅读规则，而且要对该游戏规则有一定的了解。对人的玩家存在的非法操作限制可能不足，需要合法操作。

debug 模式没有单独生成可执行文件，在 vs 上自主打开。是简单的对部分函数测试。

文件生成用 Debug 和 Release 两种模式下生成解决方案，可执行文件路径 chess\x64\Release\chess.exe,不过该文件已经复制到压缩包内，不用通过该路径也能找到。由于代码中调用图片没有写绝对路径，图片文件复制了三次，分别放在 chess 文件夹，Debug 文件夹和 Release

文件夹，因此运行时不要脱离该三个文件夹，否则图片无法调用，程序运行出现问题。由于.exe 文件属性，本程序**可能**只能在 Windows 系统运行。

移动时产生足迹，虽然有些混乱，但以始终带数字的视为棋子即可。

游戏主体几乎参照要求，在 `mode1` 函数中，在坐标初始化后，大框架 `while` 循环判断是否有一方的剩余没到终点的棋子到 0，内部分为三个玩家切换，每个玩家棋子移动分成四部分：抛出色子点数不为 6 且没有棋子出发、抛出色子点数为 6 且机场仍有棋子、抛出色子点数不为 6 且机场仍有棋子、机场没有棋子。每个玩家之间切换前加入不同颜色棋子处于同格子的判断，用于碰撞。

## （二）可能的拓展功能

简化的棋盘用图形构成要素绘制；使用 GUI 图形；色子动画用的图片切换插入及 `Sleep` 函数控制；棋子和格子结构体；棋盘格子较大，可同时容纳四枚棋子，叠子情况比较容易看。

## （三）可能的不足

可执行文件兼容系统有限；对坐标没有全局初始化，部分功能无法做成函数，代码冗余；三维数组代表所有棋子的绝对位置，较为麻烦；玩家数量和玩法选择较为固定；