

五子棋程序设计

姜俊彦¹

目录

1	实验题目	2
2	实验概况	2
3	实验过程	2
3.1	实验平台	2
3.2	实验程序	2
3.2.1	整体架构	2
3.2.2	main.c	3
3.2.3	chessboard.c	4
3.2.4	function.c	4
3.2.5	inputprocess.c	4
3.2.6	state.c	4

¹姜俊彦，中国科学院大学，2022K8009970011，jiangjunyan22@mailsucas.ac.cn

1 实验题目

五子棋程序设计：要求实现包含以下功能的五子棋程序

- 人人对战
 - 棋盘与落子显示
 - 键盘输入落子与异常输入检测
 - 模式选择与退出
 - 裁判与禁手
- 人机对战
 - 显示人机落子
 - 五子棋程序比赛

2 实验概况

本实验为来到中国科学大学，就读于网络空间安全专业的第三个系统性实验（前两次分别为：计算机科学技术导论动态网页设计与数字电路 AX7035 FPGA 开发板电路设计），也可以说是对于 C 语言学习的总结与收官。本实验自 2023 年 12 月 9 日项目创立至今（2024 年 1 月 20 日），共 43 日。本人于此期间基本完成了实验所要求的的全部内容。接下来是对程序本身及程序编写过程中的收获与反思。

3 实验过程

3.1 实验平台

- 操作系统：Windows 11 专业版 22H2, Ubuntu 22.04(64-bit)
- 编辑平台：Visual Studio Code 1.85.2
- 编译环境：gcc version 11.4.0
- 代码管理平台：GitHub

3.2 实验程序

限于篇幅这里不再粘贴大段源码，有意者可于日后前往我的 GitHub 仓库查看¹

3.2.1 整体架构

本程序大致按照以下几个主要模块（文件）架构组成。

- 棋盘、落子显示-chessBoard.c
- 常用函数-functions.c
- 输入信息接受与处理-inputprocess.c

¹<https://github.com/jiuhao47>

- 状态控制与信息显示-state.c
- 裁判 (含胜负及禁手) -judge.c
- 评分函数-judge_score.c
- 树架构 AI-ai.c
- 主函数及头文件-main.c&myhead.h

接下来将对每个部分分块介绍主要功能及其关联

3.2.2 main.c

鉴于是最短的文件,就将其整个粘贴进来了。这样做的思路来源其实是从《Python 编程从入门到实践》这本书中获得的¹

```
1  #include "head.h"
2  int main()
3  {
4      init_state();
5      input();
6      update();
7      while (gamestates.runningstate)
8      {
9          update();
10     }
11     return 0;
12 }
```

除必要的的初始化外,程序全部由 gamestates 结构体中的成员 runningstate 控制程序运行走向。而具体的程序内容则被封装在 update() 函数中,下为 update()²函数的片段,其主体是以状态机为核心搭建的,不同的状态导向不同的控制流。

```
1  //...
2  if (gamestates.runningstate == -2)
3  {
4      gamestates.runningstate = 0;
5      return;
6  }
7  else if (gamestates.runningstate == 1)
8  {
9      updateHumanInput();
10 }
11 //...
```

¹本人于 2023 年暑假期间对照此书学习了一点 Python 编程的知识,其中有一个章节介绍了游戏编程过程中的面向对象思想,而正是此影响着我的五子棋程序编写

²位于 chessboard.c 中

3.2.3 chessboard.c

棋盘、落子显示位于此模块内。主要是利用双循环进行复位、更新、打印等。此过程中更复杂的其实是对于中文编码¹的理解

```
1  for (i = 0; i < SIZE; i++)
2  {
3      for (j = 0; j < SIZE; j++)
4      {
5          ...//大量的双循环结构
6      }
7  }
```

3.2.4 function.c

一些常用的函数包，比如读取一行内容、坐标转换、乘方、最值、绝对值等

- int mygetline(char s[],int lim)
- int pointInBoard(int tempx,int tempy)
- int displayPosToInnerPos(int tempx,int tempy)
- int mypow(int x, int n)
- int weight(int x,int y)
- int mymax(int x,int y)
- int myabs(int x)

3.2.5 inputprocess.c

该模块重在对于输入内容的处理，这里需要考虑到不同情况的输入（坐标还是模式）。由于事先设计了状态机，故此处也使用了状态机的错误状态来处理非法输入（ErrorHandle() 函数）。无太多亮点不多赘述。

3.2.6 state.c

对于整个五子棋程序可能相对设计的比较精巧的就是 state.c 这个模块，gamestates 为全局变量。所有函数可见，其作为调控整个程序运行的状态机控制程序的运行逻辑

-
-
-

¹GBK 与 UTF-8 的占字节大小不同