华南农业大学课程设计报告

课程名称: 程序设计与算法课程设计

项目性质: 综合性、设计性实验

开设时间: 2020-2021 学年第二学期

姓 名: 古卓恒

学 号: 202014210104

指导教师: 梁云

目 录

第一章 课题选择	1
1.1 题目描述	1
1.2 题目要求	1
第二章 问题描述及功能叙述	2
2.1 问题描述与理解	2
2.2 功能叙述	3
2.2.1 功能 (1): 用户注册	3
2.2.2 功能 (2): 用户登录	4
2.2.3 功能 (3): 修改用户信息	4
2.2.4 功能 (4): 更新数据	5
2.2.5 功能 (5): 重置战绩	6
2.2.6 功能 (6): 绘制任意大小的棋盘	6
2.2.7 功能 (7): 用户下棋	8
2.2.8 功能 (8): 人机对战	9
2.2.9 功能 (9): 人人对战	10
2.2.10 功能(10): 查看排名	10
第三章 数据结构组织情况	12
3.1 结构体数组与哈希表	12
第四章 算法描述	13
4.1 排行榜排序算法	13
4.2 五子棋游戏算法	13
4.2.1 判断是否胜利算法	13
4.2.2 点位价值判断	13
第五章 效果与测试情况	16
第六章 总结与展望	22
6.1 总结	22
6.1.1 本游戏程序特点	22
6.1.2 本游戏程序不足	23
6.2 未来工作展望	23
参考文献	25
致谢	26

第一章 课题选择

本人选择将制作五子棋程序作为本次《高级语言程序设计》的课题

1.1 题目描述

设计人机交互下五子连棋的程序,该程序具有以下功能:

- (1) 用户注册: 注册新用户。
- (2) 用户登录:通过输入账号、密码、验证码登录系统。登录后可以得到棋盘大小等信息。
 - (3) 修改用户信息: 用户登录后可以修改用户信息和棋盘大小。
 - (4) 人机下棋。

1.2 题目要求

- (1) 按照分析、设计、编码、调试和测试过程完成应用程序;
- (2) 学习并使用流程图等工具,并在撰写报告中使用;
- (3)程序的各项功能在程序运行时,以菜单方式选择并执行;
- (4)要求用户输入数据时,要给出清晰、明确的提示,包括:输入数据的内容、格式及结束方式等
 - (5) 所有的信息存储在文件中,并实现文件读写操作。

第二章 问题描述及功能叙述

2.1 问题描述与理解

五子棋游戏是一种两人对弈的纯策略型棋类游戏。五子棋不仅能增强思维能力,提高智力,而且变化多端,非常富有趣味性和消遣性,因此为人民群众所喜闻乐见。该游戏的规则简单,即双方分别使用黑白两色的棋子,下在棋盘直线与横线的交叉点上,先形成五子连线者获胜。以下是我对该游戏程序要实现的功能的理解。

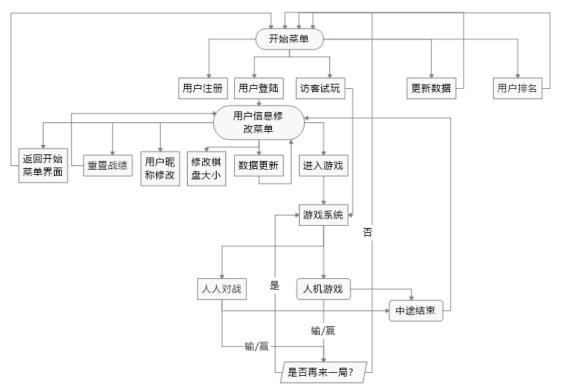


图 2-1 游戏程序基本框架

2.2 功能叙述

2.2.1 功能 (1): 用户注册

图 2-1

用户在界面通过指引输入数字到达注册界面,程序自动生成账号,用户需根据页面要求自行输入合法的用户名、密码,注册成为新用户。其中,为了确保密码的准确性和安全性,用户需要输入两次密码以确保输入正确无误。

2.2.2 功能 (2): 用户登录

```
int
                      //用户登录
 99
     Login(int r_n)
100 - {
101
         int id,N,choice;
         char p[20];
printf("请输入账户: "); scanf("%d",&id); getchar();
102
103
         N = id-32300;
104
105
         printf("请输入密码(密码小于20位): ");
         scanf("%s",p);
106
         if(!strcmp(user[N].password,p)){
107 📮
108
109
         Changeinformation(N,r_n);
110 | }
111 📮
         else{
             printf("登陆失败!"); return ERROR;
112
113
114
115
          return OK;
116
```

图 2-3

用户在界面通过指引输入数字到达登录界面,通过输入账号、密码登录系统。登录后可以得到棋盘大小、战绩等信息。

2.2.3 功能 (3): 修改用户信息

```
<u>void C</u>hangeinformation(int idpass,int r_n)  //修改用户个人信息
202 -
           system("cls");
printf("\n\n\n\n");
203
204
205
           printf("\t\t
printf("\t\t
                                * * * * * * 用户信息修改界面
206
207
           printf("\t\t
printf("\t\t
208
                                                       棋盘大小: %2d
209
                                   用户: %10s
                                                                              *\n",user[idpass].name,user[idpass].x)
210
211
                                  1.修改棋盘大小
212
                                                             2.用户名修改
214
                                   3.数据更新
                                                                4.进入游戏
216
217
218
                                  编写者: 古卓恒 20网工1班 202014210104
219
           printf("\t\t * printf("\t\t * * * * * * * printf("请输入数字: ");
220
221
222
223
224
           int choice;
scanf("%d",&choice);
225
           while(choice!=1&&choice!=2&&choice!=3&&choice!=4)
227
               printf("非法输入, 请重新输入: ");
scanf("%d",&choice);
228
229
230
```

```
if(choice==1){
232
233
                  modification(idpass);//查看棋盘大小、战绩等信息。提供修改棋盘大小选项
234
235
              else if(choice==2)
236
                      printf("请输入新的用户名:");
237
238
                      getchar();
                      gets_s(user[idpass].name); //修改用户名
printf("%s,修改用户名成功! \n",user[idpass].name);
system("PAUSE");
239
240
241
242
                      Changeinformation(idpass,r_n);
243
244
              else if(choice==3)
245
246
                  //数据更新
247
              else if(choice==4)
248
249 📮
250
                 gamecontrol(idpass,r_n);
251
252
              getchar();
printf("\n按Enter键进入游戏!");
253
254
              char Enter;
255
              Enter = getchar();
              while(Enter!=10)
256
257
258
               Enter = getchar();
259
260
              if(Enter==10)
261
262
                system("cls");
                gamecontrol(idpass,r_n);}
263
264
```

用户在界面通过指引输入数字到达登录界面,登录后可以修改用户信息和棋盘大 小。

2.2.4 功能 (4): 更新数据

用户信息经过一系列操作后会发生改变,通过更新数据选项可以更新所有账户的用户信息,并写入txt文件中。

2.2.5 功能 (5): 重置战绩

用户决定从头来过东山再起了可以选择重置战绩。

2.2.6 功能 (6): 绘制任意大小的棋盘

```
212 void draw(int x)
213 📮 {
214
          int i,j,k,q;
          int a, b, c;
char*** p = new char** [x];
215
216
                                                //动态分配构造三维数组,用于存储该位置图像
217
          for (a = 0; a < x; a++)
218
219
              p[a] = new char* [x];
220
              for (b = 0; b < x; b++)
221
222
                   p[a][b] = new char[x];
223
          for (a = 0; a < x; a++)
224
225
226
227
              for (b = 0; b < 4; b++)
228
229
                   for (c = 0; c < 4; c++)
230
231
                       p[a][b][c] = 1;
232
233
234
235
          for(j=0;j<x;j++)
236
              for(i=0;i<x;i++)
237
                  if(board[j][i]==SPA)
strcpy(p[j][i]," \0");
if(board[j][i]==MAN)
strcpy(p[j][i]," \0");
if(board[j][i]==COM)
strcpy(p[j][i]," \0");
238
                                                 //若为空,则记录空
239
240
241
                                                 //若人在此下棋,则记录 • \0
242
243
                                                 //若电脑在此下棋,则记录 ◎ \0
        printf("\n
                                     下子请输入坐标x y(Enter键输入结束)
                                                                           悔棋请输入66 66 \n\n");
```

```
246
247
         printf(" 0");
for(k=1;k<x-1;k++){
   printf(" %2d",k);</pre>
248 –
249
                                             //写下横坐标
250
         printf(" %2d \n",x-1);
251
252
253
             printf("
254
         255
                                             //接着绘制中间的线
256
257
         printf("- - - 1 - - >x\n");
258
                                             //第一行的尾长这样,因为特殊,特殊处理绘制下来
259
         for(i = 0; i<x-1 ; i++){
    printf(" %2d| ",i);
    for(j = 0;j<x-1 ; j++){
        printf("%3s| ",p[i][j]);
}</pre>
                                             //绘制棋子
//先写下纵坐标
260
261
262
263
264
265
             printf("%3s| %d\n",p[i][j],i);//绘制最后一个棋子,并再写一次纵坐标,方便查看棋盘
267
                                             //棋子下面的线 头长这样,因为特殊,特殊处理绘制下来
                 for(j = 1;j<x-1 ; j++){
    printf("+ - - - ");
268
269
                                             //接着绘制中间的线
270
             printf(" + - - - | \n");
                                             //棋子下面的线 尾长这样,因为特殊,特殊处理绘制下来
```

```
//绘制最后的线
printf(" %2d| ",i);
for(j=0; j< x-1; j++){
    printf("%3s| ",p[i][j]);
275
276
277 -
278
279
                   printf("%3s| %d\n",p[i][j],i);
280
281
             282
283 📥
284
285
286
             printf("- - - J \n");
printf(" 0");
for(k=1;k<x-1;k++){
    printf(" %2d",k);</pre>
287
288
289
290
              printf(" %2d \n",x-1);
291
292
```

程序通过用户传入的合法的棋盘信息生成合法的、图像化的棋盘,并在棋盘外标上坐标,以方便用户下棋。

本人绘制棋盘的思路是一行接着一行的画,具体步骤如上图代码所示,其中注释已说明详尽。

2.2.7 功能 (7): 用户下棋

```
86
                      printf("玩家下在%d %d\n",X,Y);
d1=Z;       /* 储存电脑上手棋型 */
 87
 88
              c1=W; d1=Z;
                   if(win(W,Z,x))/* 电脑赢 */
 89
 90 =
                       printf("要悔棋吗? 请输入y或n: ");
                       getchar();
scanf("%c",&ch);
while(ch!='y'&&ch!='n')
 92
 94
 95 🗀
                            printf("输入错误,请重新输入:");
 96
                            scanf("%c",&ch);
 97
 98
 99
                       if(ch=='n')
100 📥
101
                            printf("电脑赢!下次继续努力! \n");
102
                           return;
103
104
                       else
105 🗀
106
                           max+=2;
107
                           board[Z][W]=SPA;
108
                           board[b1][a1]=SPA;
                       system("cls"); draw(x);
} /* 悔棋 */
109
110
111
                   printf("电脑下在%d %d\n请输入: ",W,Z);
scanf("%d%d",&X,&Y); /* 玩家下子 */
if(Y==66&&X==66) /* 悔棋 */
112
113
114
115 🖶
                       max+=2;
117
                       board[Z][W]=SPA;
                       board[b1][a1]=SPA;
118
                       system("cls");
119
120
                       draw(x);
122
                       scanf("%d%d",&x,&Y);
123
                  while((X<0||Y>x-1)||(Y<0||Y>x-1)||board[Y][X]!=SPA)
124
125 🗀
126
                       printf("坐标错误或该位置已有子!请重新输入:");
127
                       scanf("%d%d",&x,&Y);
128
129
                  a1=X; b1=Y; max--;
130
                  board[Y][X]=MAN;
131
                  system("cls");
132
                  draw(x);
133
                  if(win(X,Y,x))
134 🗀
135
                       printf("你真棒! 电脑输!!!\n");
136
                       user[N].wins++;
                       printf("胜场为: %d",user[N].wins);
138
                       return;
139
                       /* 玩家赢 */
140
141
              printf("和局\n");
142
```

用户在界面输入棋盘坐标,程序判断坐标合法,则成功下棋;若坐标在界外,则 提示玩家该坐标在界外,并要求重新输入;若坐标非空,则提示玩家坐标非空,并要 求重新输入。玩家输入 DONE,则以平局结果退出游戏。

2.2.8 功能 (8): 人机对战

```
if(choice==1)
             /* 人机模式 */
42 📮
            43
44
45
46
47
48
        uraw(X),
printf("\n\t\t\t先手请按1,后手请按2:");
int z; //先下后下选择
scanf("%d",&z);
while(z!=1&&z!=2)
49
50
51
52
53 🖶
54
            printf("输入错误,请重新输入:");
            scanf("%d",&z);
55
56
         if(z == 1)
                                     //先手
57
58 🚊
59
            printf("请下子:");
60
         scanf("%d%d",&X,&Y);
while((X<0||X>x-1)||(Y<0||Y>x-1)){
    printf("坐标非法!请重新输入: ");
    scanf("%d%d",&X,&Y);
61
62 📥
63
64
65
66
         a1=X;b1=Y;max--;
67
         board[b1][a1] = MAN;
                                     //棋盘
         system("cls");
68
69
         draw(x);
70
71
         while(max!=0)
72 📮
73
              if(max == x*x)
74 🖶
75
                       W=x/2; Z=x/2;
76
                       board[Z][W]=COM; max--;
77
                       system("cls");
78
                       draw(x);
79
                      /* 电脑先下就下在7 7 */
80
              else
81 =
                                                   //电脑下棋算法
82
                       AI(&W,&Z,x);
                       board[Z][W]=COM; max--;
83
                       system("cls"); draw(x);//画下电脑下棋的操作
电脑下子 */
84
                     /* 电脑下
```

```
printf("玩家下在%d %d\n",X,Y);
=Z;       /* 储存电脑上手棋型 */
87
88
            c1=W; d1=Z;
89
                if(win(W,Z,x))/* 电脑赢 */
90 🖶
                    printf("要悔棋吗?请输入y或n:");
91
92
                    getchar();
                    scanf("%c",&ch);
93
94
                    while(ch!='y'&&ch!='n')
95 🖶
                        printf("输入错误,请重新输入:");
96
97
                        scanf("%c",&ch);
98
99
                    if(ch=='n')
100 草
101
                        printf("电脑赢!下次继续努力! \n");
102
                        return;
103
104
                    else
105 🛱
106
                        max+=2;
                        board[Z][W]=SPA;
107
108
                        board[b1][a1]=SPA;
109
                         system("cls"); draw(x);
```

程序通过算法实现人机交互下游戏的各种难度,随着用户的胜场次数增加,游戏难度由 易到难线性递增。

2.2.9 功能 (9): 人人对战

程序可以实现用户与用户间对战。

2.2.10 功能 (10): 查看排名

```
void
                                            //数据结构排序方法
322
      CheckRank(int r_n)
323 📮
         system("cls");
printf("\t
324
                                        * * * 排行榜 * *
325
326
         struct RankQueue
327
328
             char name[10000];
329
             int wins;
330
             int ranks;
331
          }rankson[r_n];
332
333
          int r = 1;
334
          for(r = 1;r<=r_n;r++)
335
336
             strcpy(rankson[r].name,user[r].name);
337
             rankson[r].wins = user[r].wins;
             rankson[r].ranks = 0;
338
339
```

```
341
           int b,u,temp;
342
           char name[10000];
343
           for(b = 0;b < r_n;b++)
344 🖶
           {
345
              for (u = 0; u < r_n-b-1; u++)
346
347
                   if(rankson[u].wins>rankson[u+1].wins)
348
349
                        temp = rankson[u].wins;
350
                        rankson[u].wins=rankson[u+1].wins;
351
                        rankson[u+1].wins=temp;
352
                        strcpy(name, rankson[u].name);
353
                        strcpy(rankson[u].name,rankson[u+1].name);
354
                        strcpy(rankson[u+1].name,name);
355
                }
356
357
358
359
      int i,z=1;
for(i = r_n-1;i >0;i--)
360
         printf("\t\t %2d* %20s
361
                                    胜场: %d\n\n\n",z,rankson[i].name,rankson[i].wins);
362
                                                                         程序显
```

示排行榜,用户可查看所有玩家的排名。

第三章 数据结构组织情况

3.1 结构体数组与哈希表

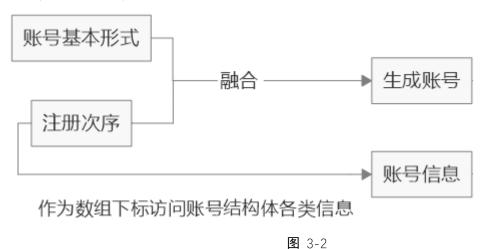
因为 C 语言并不支持面对对象,所以我们需要建立结构体来存储账号、账号名、账号密码等账号基本信息。同时,账号的访问追求的是速度,尤其是在账号量过多的情况下,所以,我建立了结构体数组来存储账号信息。

```
//个人账户
    /*typedef */struct PersonalAccount
24 ☐ {
25
     int ID;
26
     char name[100000];
     char password[10000];
27
                                 //胜场
28
     int wins:
29
     int rank;
     int x;
                               //棋盘x轴
30
    } user[MAX];
```

图 3-1

账号信息的访问采用建立散列表(Hash Table)的思想,即根据关键码值(Key value)而直接访问账户相关信息的数据结构。

具体表述为:参考 QQ 的账号建立,账号由 QQ 的服务器自动生成,本程序的账号基本形式在代码中写出,账号的生成由注册次序加账号基本形式决定,在调用各账号时,注册次序为账号数组的位置。



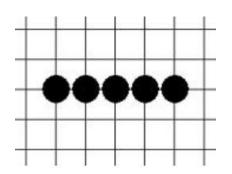
第四章 算法描述

4.1 排行榜排序算法

排行榜的建立需要依托于算法的实现。在本程序中,本人在排行榜的算法上采用了冒泡排序法。冒泡排序是一种简单的排序算法,它也是一种稳定排序算法。其实现原理是重复扫描待排序序列,并比较每一对相邻的元素,当该对元素顺序不正确时进行交换。一直重复这个过程,直到没有任何两个相邻元素可以交换,就表明完成了排序。该算法的最差时间复杂度为 O(n^2),最优时间复杂度为 O(n),平均时间复杂度为 O(n^2)。

4.2 五子棋游戏算法

4.2.1 判断是否胜利算法

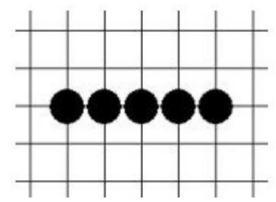


首先,我们需要先明确定义,在五子棋中,一个子共有八个方向可以连为五子,连 五而游戏胜利。

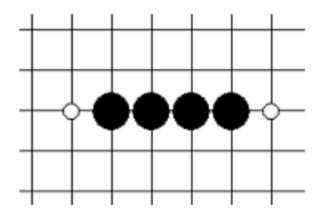
因此,我们需要对每个子遍历它的八个方向,若在某一方向的下一个点位发现状态相同,则沿着此方向继续观察下一点位。当五个状态相同时,则结束遍历,返回胜利判断,否则,跳出这一方向判断,遍历下一方向,若八个方向都遍历完发现没有连五子情况,则结束循环,返回失败判断。

4.2.2 点位价值判断

在研究此算法前,我们首先需要了解五子棋的各种情况和定义。 最常见的基本棋型大体有以下几种:连五,活四,冲四,活三,眠三,活二,眠二。 ① 连五: 顾名思义, 五颗同色棋子连在一起, 不需要多讲。

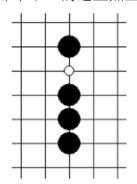


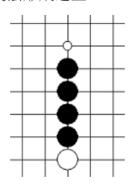
② 活四: 有两个连五点(即有两个点可以形成五),图中白点即为连五点。

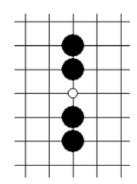


③ 冲四:有一个连五点,如下面三图,均为冲四棋型。图中白点为连五点。

相对比活四来说,冲四的威胁性就小了很多,因为这个时候,对方只要跟着防守在那个唯一的连五点上,冲四就没法形成连五。

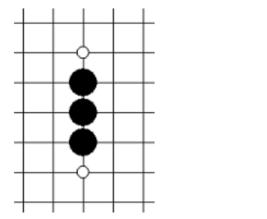


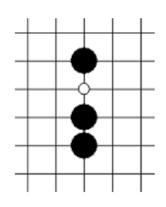




④ **活三**:可以形成活四的三,如下图,代表两种最基本的活三棋型。图中白点为活四点。

活三棋型是我们进攻中最常见的一种,因为活三之后,如果对方不以理会,将可以下一手将活三变成活四,而我们知道活四是已经无法单纯防守住了。所以,当我们面对活三的时候,需要非常谨慎对待。在自己没有更好的进攻手段的情况下,需要对其进行防守,以防止其形成可怕的活四棋型。





其它情况因研究不充分,在此不一一赘叙。

介绍完五子棋的各种情况和定义后,下面我将对算法进行更为详尽的介绍。

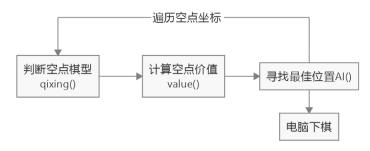
本 AI 算法的基本思路是: 遍历所有空点,找到价值最大的点,并在该点下棋。 其中,缩小遍历范围和价值判断规则是使该算法更加快速和智能的关键,本节将对 这两大难点着重介绍。

缩小遍历范围是提升算法效率的关键。一个标准五子棋棋盘的大小为 15 * 15, 因此每次下子最多可能有 200 多种下子的方法,而本程序可根据用户要求随意修改 棋盘大小,所以下子方法将会更加庞大,在搜索的时候依次考虑每一种走法,会大大 影响算法的效率。所以我们还需适当缩小遍历范围,以加快算法速度。

我的解决方案是:记录最 X, Y 最大值和最小值,并对这个边界进行一定程度的拓展。根据下子的经验,下子只要考虑与某个棋子距离小于等于 2 的位置即可,所以,我以 X, Y 最大值和最小值为边线,向外拓展两列和两行,这样就使得算法的速度加快。

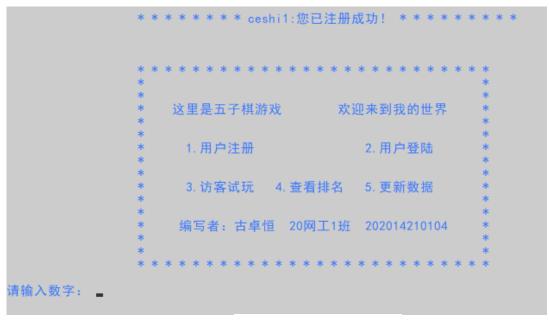
价值判断规则是提高算法智能的关键。五子棋的规则虽然简单,但下法多样,算法的全面性至关重要,这就对算法编写者提出了很高的技术要求。

我的解决方案是:首先,对五子棋的规则进行了一定的研究,并着重研究了常见的情况,如:己活、己冲,对活、对冲、活四、活三、冲三等。接着,编写 qixing()函数,判断不同的棋形,并对不同的棋形,根据己方、对方等情况进行编号。最终定义出所有的棋型的编号,并对每种情况的棋形进行赋值,并返回值 k。其次,编写 value()函数,用数组存储下所有棋型的值,根据 qixing()函数返回的值进行赋值,返回值 value。最终,找到 value 最大的位置,并在该点下棋。



第五章 效果与测试情况

游戏初始界面



注册成功

■ accountbook - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

32301 1234 123456 9 0 0

32302 a 123456 0 0 8

32303 wilgu 123456 10 0 0

32304 williams 123456 100 0 0

32305 williams gu 123456 7 0 0

32306 gu zhuo heng 17702611689 3 0 0

32307 ceshi mima1234 0 0 0

32308 ceshi1 ceshimima 0 0 0

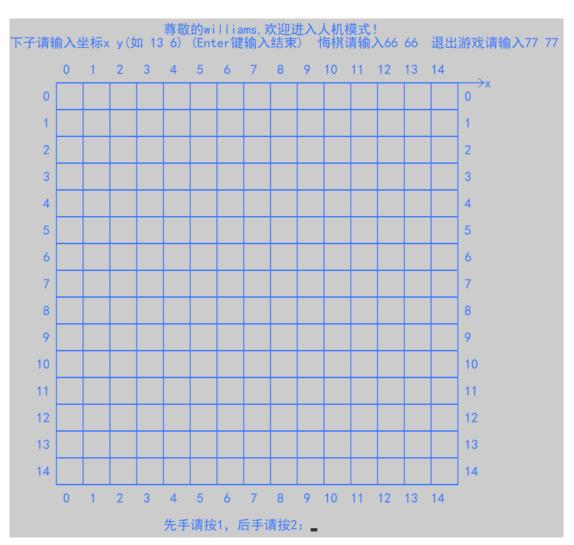
信息写出本

* * * * * *	* * * * 排行榜 * * * * * * * williams	* * * * * * * * 胜场: 100
2*	wilgu	胜场: 10
3*	1234	胜场: 9
4*	williams_gu	胜场: 7
5*	gu_zhuo_heng	胜场: 3
6*	1234	胜场: 0
7*	xiaogu	胜场: 0
8*	ceshi1	胜场: 0
9*	ceshi	胜场: 0
10*	a	胜场: 0

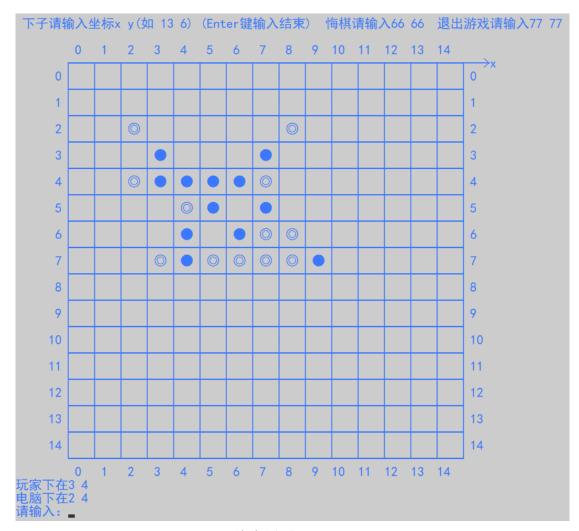
用户排名

用户注册界面

用户信息修改界面和功能展示



下棋棋盘



游戏展示

第六章 总结与展望

6.1 总结

6.1.1 本游戏程序特点

- (1) 该程序实现了信息管理系统与五子棋游戏的充分结合。本实验拥有完整的登录、注册、查看详细信息、修改用户个人信息功能,基本实现了市面上交互软件的交互功能;五子棋游戏有先手后手的选择、悔棋选项、下棋选项,完全实现了五子棋游戏的各种功能。
- (2) 该程序在所有界面下都考虑了各种情况,设置了完备的回路。
- (3) 信息管理系统设计了安全检验环节。设计的安全检验环节确保了程序的安全与数据文件的安全。
- (4) 该程序源代码的功能分区清晰完善。整个大型程序被分为三个文件,两个自建编程者头文件和一个 main 文件,两个自建头文件分别写下了五子棋游戏相应函数和信息管理系统相应函数,main 文件则负责串联两个板块的函数,其中配有充足的注释,使得程序易于查看、理解和完善。同时,自建库编写了大量实用性广泛的函数,函数的功能可以通过读取函数名称和函数注释而理解。同时,在下次编写程序时可以进行调动,缩短了接下来程序程序的时间。
- (5) 该程序可实现数据的交互。该程序可以将用户数据从内存中写出,并在下一次运行后写入,运行中途可手动将数据从内存中写出,实现了数据流的半自动化的持续性传递。
- (6) 五子棋计算机算法比较智能。程序会根据对手下的棋位遍历剩余所有棋盘的空的 位置,并寻找出价值最大的空位下子,实现了下子的智能化。
- (7) 设计程序的过程查阅了大量的文献资料。程序设计的大部分内容和思想都不能在上学期的《高级语言程序设计》中学到,在完成课程设计的过程中,需要强大的自学能力。本人在设计程序的过程中,去多方渠道了解大量材料,学习相关知识,最终确保了了课程设计课程的完成和自身能力的提升。

6.1.2 本游戏程序不足

- (1) 游戏程序设计的研究方向从开始就选择错误。刚开始在选择"五子棋"项目时,在当初现有的知识框架中,没有图形库的概念,所以从游戏设计之初就选择了比较简单的命令行输出模式,而当看到同学用图形库设计出较为现代化的游戏时,我已经没有时间和精力再更改自己的研究方向,所以最终设计出的游戏程序的执行依托于命令窗格,整个游戏虽然实现了完整的信息管理系统功能和五子棋游戏功能,但整体感观较为原始和简单,不能追上现代化游戏程序设计的步伐。
- (2) 五子棋算法不够完善。本人由于时间和能力有限,不能在五子棋的算法上进行深入的研究和探讨,所以没有实现计算机算法难度随下棋者胜场率提升而增强和闯关功能,使得游戏算法比较单薄,这是我觉得非常遗憾的。
- (3) 游戏程序的闭环可能存在问题。尽管本人对程序所有过程经过多次的测试与校验,但因为游戏程序系本人独立完成,所以在测试与校验的过程中可能存在逻辑缺陷。 且该程序未经他人大量的使用、测试与校验,因此该程序可能难免会存在问题或 bug。
- (4) 本人自控能力和自学能力亟待提升。在游戏设计上,有一个不可否认的事实就是: C++因为属于面对对象的语言,所以用 C++进行游戏设计相比于用 C 语言更加的好。但本人自控能力还不够强大,寒假也没有怎么打代码,更不用说学习新的语言了,所以在游戏程序设计的语言的选择上没有选择 C++,导致游戏程序的相对低级化和简单化。

6.2 未来工作展望

- (1) 在课程设计完成后,加强自控能力与自学能力,学习 Python 和 C++等语言,加深对各种库、模块的理解和学习。同时,将所学知识与本人的网络工程专业结合,探索 web 领域和网络工程领域,争取在暑假设计出一个现代化的、功能完备的、实现鼠标点击交互的、图形化的、网络化的、体验舒适的游戏程序。
- (2) 继续加强对各类算法的学习。算法是编程能力的基础,编写算法能力的提升代表 着编写代码能力的提升,在下一阶段,本人要加强对五子棋规则和各种五子棋算 法的理解和学习,使得本程序的人机对战模块更加丰满以及具阶梯性。
- (3) 加强对代码的掌握。"书到用时方恨少",这句话用在计算机领域也并无错处,制作大型项目,对各个方向的了解与掌握,对各类库、模块、函数的理解和运用,甚至对基础语法的掌握程度,都会影响项目的编写速度和最终的功能展示。在下一阶段,本人要加强对代码的掌握程度,丰富自己的知识深度与知识广度。

(4) 开始着手了解计算机的前沿领域,并开始对各种深度学习模型的结构的理解与学习。

算法自编写下来之后,就无法根据对象进行自我的学习和提升。而深度学习则可以不断学习完善模型,这是计算机的未来和造福人类的技术。深度学习模型结构貌似复杂,但其实早以实现了权重轻量化、精度高、速度快等特点,例如最新的YoloV5模型,权重大小只有27M,但识别精度高、运行速度快,目前具有广泛的运用前景,Yolo的其它旧的模型已经与计算机视觉等结合并运用于异物检测等方向。这是计算机领域的前沿科学,也是本人的心之所向,在下一阶段,本人要着手研究这一领域,并尝试着运用这些技术做出一些项目和成果。

以下是学习 C 语言基础语法的一些摘抄:

大型程序设计

(1) 避免赘余,建立个人库:头文件

函数声明中使用关键字 extern 是通知编译器该函数的定义会提供给链接器 extern void

print_plant();

#include<>:告诉预处理器头文件中可以在系统目录中找到,

#include"":将它标记为有关属于编程者库的信息

在解决方案的两个分离部分的公共边界称为:接口

头文件的目的就是定义库和该库程序间的接口

(2) 数据类型

C语言有五种数据类型

auto 函数参数和局部变量的默认存储类型;在函数调用时自动分配存储空间到 栈上,在函数返回时释放存储空间

extern 链接器所知名称的存储类型

static 在程序执行之前只分配一次存储空间的变量的存储类型,该变量会在多次函数调用之间保留。

register 程序员要存储在寄存器中的自动变量的存储类型

(3) 条件编译

#if defined (TRACE)

printf("");

#endif

如果 TRACE 在预处理中定义,那么执行打印,否则 TRACE = 0,类似开关操作 defined 运算符只能在#if 和#elif 间存在

参考文献

- [1] Problem Solving and Program Design in C, Sixth Edition, Jeri R.Hanly, Elliot B.Koffman
- [2] https://blog.csdn.net/weixin_44062380/article/details/105881036, c++ 五子棋 AI 算法及实现 详细解析
- [3] https://blog.csdn.net/qq_44732921/article/details/102620510, 五子棋 AI 算法
- [4] 严蔚敏, 李冬梅, 吴伟民, 数据结构(C语言版)(第二版)
- [5] 朱全民, 陈松乔. 五子棋算法的研究与思考[J]. 计算技术与自动化, 2006, 25(002):71-74.
- [6] 张海峰, 白振兴, 张登福. 五子棋中的博弈智能设计[J]. 现代电子技术, 2004, 027(007):25-27.
- [7] 陈雪荣,岳书丹. 基于 Java 的五子棋游戏的设计与实现[J]. 信息系统工程, 2020, No.320(08):106-107.
- [8] 程宇, 雷小锋. 五子棋中 Alpha-Beta 搜索算法的研究与改进[J]. 计算机工程, 2012.
- [9] http://game.onegreen.net/wzq/HTML/142336.html, 五子棋基本棋型及其特点
- [10]https://blog.csdn.net/weixin_44369212/article/details/89058548, 学生信息管理系统——C语言版
- [11]https://blog.csdn.net/NGUP_LEE/article/details/106965990, C++学生信息管理系统(完整版)
- [12]https://blog.csdn.net/qq_44002167/article/details/89087778, C 语言实现学生信息管理系统
- [13] https://blog.csdn.net/xyh_Adolph/article/details/26730071, 五子棋 AI 博弈树之带 Alpha-Beta 剪枝的极大极小过程函数

致谢

光阴似箭,我即将在华南农业大学度过自己的十八周岁,为期半学期的《高级语言程序设计课程设计》课程也即将结束。在这一过程中,我从测绘工程专业转入网络工程专业,面对新的同学,面对新的老师,面对新的学科,在感受着新鲜美丽、青春洋溢的大学生活的同时,也是顶着更大的压力不断前行。

在这一过程中,感谢每个老师无私的帮助和教授,梁云老师在第一次代码检查时一针见血的指出了我的代码不足和问题,让我在之后的代码编写中有了更加明确的改进需求和努力方向;感谢在一起奋斗过的战友,尤其是与我一同转专业过来的詹煌超同学,你高超的技术和上进的心激励着我不断进步,你的幽默使得冰冷的课程变得有趣,你的无私帮助给了我坚实的支持;感谢网工1班的班委和同学,是你们的无私体贴和帮助让我更加快速的适应了转专业后的新生活。在此不能一一致谢,深表歉意。

在完成课设的过程中,提升的自学能力,掌握的查阅资料、文献能力,得到的积极 探索的精神、永不言弃的毅力,是我人生中的宝贵财富,再次感谢老师的悉心指导和良 苦用心!