

“程序设计基础” 课程设计

程建军

2020 年 12 月 6 日

内 容

1 题 目	1
1.1 通讯录	1
1.2 动物棋	3
2 总体要求	5
3 期望	5

I 题 目

1.1 通讯录

设计并编写一个系统，对用户的通讯录进行管理，提供以下功能：

1. 分组：通讯录中的信息按分组进行管理，如将联系人分为家人、好友、同学、生意伙伴、普通朋友等多个组，组名由用户输入；
2. 联系人：通讯录中的每位联系人的信息包括其姓名、性别、籍贯、生日、联系电话（手机、固定电话）、QQ 号、Email、家庭住址、组别等信息 (可以自己添加相关信息)；
3. 界面：程序运行时显示如下界面 (主界面)
 1. 录入信息
 2. 查看信息
 3. 修改信息
 4. 删除信息请选择 (q: 退出):

4. 录入信息：主界面显示时，用户输入 1 表示录入联系人或组信息，此时显示如下选项：

1. 新建组

2. 新建联系人

请选择 (q: 返回主界面)

如用户要新建一个组，则输入 1，程序给出输入组名的提示，用户输入组名即可，如果用户输入的组名不存在，则新建一个组，如已经存在，则给出提示；输入 2 表示用户要录入一个新的联系人的信息，若输入的组名是已经存在的组，则表示将该联系人放在该组中，若输入的组名不存在，则表示新建一个组并将该用户放入该组中；

5. 查看联系人信息：主界面显示时，输入 2 查看联系人的信息，可以按组查看，显示该组中所有联系人的信息，也可以按联系人的姓名或手机号码查看，显示所有该姓名或该手机号的联系人信息；

6. 修改信息：在查看时发现录入的联系人信息有误，则可以在主界面显示时输入 3，接着输入联系人的姓名等信息，表示要更改该联系人的信息，然后输入正确的信息；

7. 删除信息：主界面显示时输入 4，接着输入联系人姓名等信息，表示要将该联系人的信息删除掉，也可以输入组名将该组删除，同时也删除该组中所有联系人；

8. 退出：主界面显示时输入 q，则程序退出执行。

1.2 动物棋

编写一个动物棋游戏，支持 A、B 双方对战，A 为计算机，B 为用户。棋盘为 8×8 的方格，如图1所示，每个方格内只能放入一枚棋子。每一枚棋子代表一只动物，涉及的动物有象 (E)、虎 (T)、

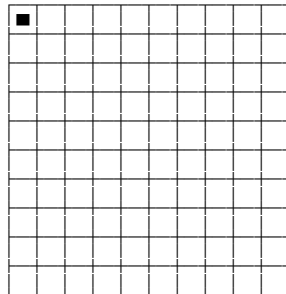


图 1: 动物棋的棋盘

狮 (L)、狼 (W)、狗 (D)、猫 (C) 和鼠 (M)，每一类动物拥有各自的等级，等级按此顺序从高到底。

游戏开始时，对战的双方各有每种动物两只；在棋盘的下方显示 B 方每种动物的数量，并将 A 方的棋子随机地排放在棋盘上某些方格内，对应的方格中显示一个 ‘●’ 字符，使 B 不知相应的方格中是何种动物。B 方可使用上、下、左、右键移动黑色方块到所需的方格中，按下代表动物的字母键将该动物落子到该方格中。如果 B 方的动物与 A 方的动物处于相邻的方格中，则高等级的动物会消灭低等级的动物，但鼠会消灭象。被消灭的动物从棋盘上消失，如果是 B 方的动物被 A 方消灭，则棋盘下方相应动物的数量被更新。B 方也可以将黑色方块移动到己方棋子所在的方格，按下空格键选中该棋子，然后使用上、下、左、右键移动到合适的方格中后，再次按下空格键将该棋子重新落子到目标方格中。以这样的方式一直进行游戏，直到一方的动物被另一方完全消灭，则还有剩余棋子的一方获胜。如果最后双方均只有一枚棋子剩余，而且是相同的动物，则平局。

功能要求

1. 实现游戏的对战功能；
2. 提供复盘功能：为了支持该功能，在游戏运行时，应将对战过程的 A、B 双方的落子情况记入文件中。程序启动时，允许用户选择是要进行游戏还是对曾经进行过的游戏进行复盘。如果用户选择的是后者，则允许用户输入记录对战过程的文件名，程序从文件中读入记录的数据，根据这些数据自动展示对战过程。

说明

1. 棋盘的绘制：本游戏是基于控制台的应用程序，棋盘可以通过输出特殊字符的组合而进行绘制。图1中的棋盘左上角是一个 ‘┐’ 字符；方格上横线、下横线中间是 ‘—’，最上面一行每一方格的右上角是 ‘└’；最右上角是 ‘┌’；每个方格的中间的竖线是 ‘|’，最左边一列方格

的左下角是 ‘└’，最右边一列方格的右下角是 ‘┘’，其余方格十字交叉的地方是 ‘+’；最左下角是 ‘┌’，最右下角是 ‘┐’，最下面一行每一方格的右下角是 ‘└’。

2. Windows 中用 `getch()` 函数，可以在键盘上按下相应的键，不需要回车就可以输入一个字符，且输入的字符不显示在屏幕上；Linux/Mac 的 `ncurses` 库中的 `getch()` 也可以起同样的作用。使用 `ncurses` 库首先需要 `#include <ncurses.h>`，然后在程序启动的时候调用 `initscr()`，就可以使用 `getch()` 不回显、不等确认地输入字符了。在程序结束之前需要调用 `endwin()` 退出 `ncurses` 库。

3. 本游戏需要在控制台窗口中定位输出字符的位置，可以用下面提供的函数实现这一功能

- Windows 版

首先 `#include <Windows.h>`，然后调用如下函数

```
void setcursorpos(unsigned x, unsigned y){
    COORD pos;
    pos.X=x;
    pos.Y=y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), pos);
}
```

- Linux/Mac 版

```
void setcursorpos(unsigned x, unsigned y){
    printf("%c[%d;%df", 0x1b, y, x);
}
```

`x`、`y` 为光标的位置，`x` 为行号，`y` 为列号；屏幕左上角为坐标原点。

4. 游戏运行时，在棋盘的方格中显示一个方块（‘□’ 字符）表示该方格被选中，按下相应棋子对应的字母键表示将该棋子落到该方格内。此时应将系统提供的光标进行隐藏，以取得较好的视觉效果。可以调用以下函数隐藏、显示控制台窗口中的光标。

- Windows

定义一个全局变量：`CONSOLE_CURSOR_INFO cci`；需要隐藏光标时调用以下函数

```
void hidecursor(){
    static char bFirst=1;
    if(bFirst){
        GetConsoleCursorInfo(GetStdHandle(STD_OUTPUT_HANDLE), &cci);
        bFirst = 0;
    }
    CONSOLE_CURSOR_INFO c=cci;
    c.bVisible = 0;
}
```

```
    SetConsoleCursorInfo(GetStdHandle(STD_OUTPUT_HANDLE), &c);
}
```

在程序结束之前调用一下函数恢复光标的显示

```
void displaycursor(){
    SetConsoleCursorInfo(GetStdHandle(STD_OUTPUT_HANDLE), &cci);
}
```

- Linux/Mac

```
void hidecursor(){
    printf("\033[?25l");
}
void displaycursor(){
    system("stty echo");
    printf("\033[?25h");
}
```

2 总体要

1. 程序结构良好，没有明显错误，界面简洁实用。
2. 准备幻灯片，对项目中自己认为技术含量最高的部分进行讲解，并阐述开发过程中遇到的难点、痛点（Pains）及如何解决的，同时总结自己的收获（Gains）——No Pain, no gain。
3. 提交的内容包括
 - (a) 课程设计报告（电子版）
 - (b) 源代码（整个项目的文件）
 最后提交时，二者缺一不可，如任何一部分缺失，则以未提交论处。
4. 题目中明确要求的功能必须实现，可能存在一些实用功能，题目中没有明确提出，大家自由发挥，朝好的方面努力！**Have a try!**

3 期望

大家充分发挥自己的主观能动性，积极动手，不拘一格，灵活地实现，**并尽可能考虑用户使用的方便!!!** 编程过程中肯定会有不少问题出现，希望大家不要气馁！遇到问题时，可以互相讨论，可以查资料，但**切记不要抄袭！**。

恳请大家认真对待本次课程设计，不要抱着完任务的态度敷衍了事！希望通过此次课程设计，让大家对软件开发有个初步的认识——有付出，一定会有回报！希望大家百尺竿头，更进一步，以此做为入门的第一步，在以后的学习、工作生活中再去体验更大的项目的开发！