

- 【注意:】** 1、本次作业只允许使用到目前为止所讲内容(第 6 章及以前)及已完成作业中的补充概念
 2、不允许使用 goto 语句
 3、在 VS2015 下做到 “0 errors, 0 warnings”

综合题 2: 彩球游戏

【游戏规则:】

- 1、游戏区域为 9*9，共有 7 种颜色的彩球随机出现，其中初始状态 5 个，以后每次出 3 个
- 2、按 F3 键可以预告下次出现的三个彩球的颜色，再按一次则关闭预告
- 3、按 F5 键可以在侧边显示当前状态统计，再按一次则关闭统计信息
- 4、用鼠标选中某个彩球，再选择一个空白区域做为目标位置，如果从源到目标位置有通路可达，则将彩球移动到目标位置；如果无通路可达，则不移动并给出相应提示
- 5、当同色彩球在横向、纵向、斜向达到 5 个及以上时，可以消除，同时得到相应的分数
- 6、按 F4 可以重新开始一局
- 7、在文件 winlines.res 中记录游戏最高得分，下次游戏得分超过最高得分的，要更新文件

【显示要求:】

- 1、被选中的彩球要有不同效果
- 2、彩球移动时，要有动画效果沿着通路进行移动
- 3、消除时要有相应的动画效果

【要求:】

- 1、所有小题放在一个程序中，用菜单方式进行选择，并加入图形化演示的要求

```

1. 内部数组，随机生成初始5个球
2. 内部数组，随机生成60%的球，寻找移动路径
3. 内部数组，完整版
4. 画出n*n的框架（无分隔线），随机显示5个球
5. 画出n*n的框架（有分隔线），随机显示5个球
6. 9*9的框架，50个球，完成一次移动
7. cmd图形界面完整版
0. 退出
-----
[请选择0-7]
  
```

- 2、为了降低难度，循序渐进，将本题分解为若干小题，完成每小题就能够取得相应的分数

菜单项 1: 输入行列后，在规定范围内随机生成五个球的位置，然后打印整个内部数组

- 为方便观察，打印时有球的位置用不同颜色输出

菜单项 2: 输入行列后，在规定范围内随机生成 60% 的球的位置，然后输入要移动球的起始坐标及目的坐标，找出将球移动过去的路径（不要求最短，能找到即可）

- 起始位置必须有球，目的位置必须为空
- 生成过程中，如果该位置已经有球，要重新生成

菜单项 3: 结合菜单项 1 和 2，完成一个完整的实现过程（纯内部数组表现形式）

- 球的位置用不同颜色标出
- 连续 5 个则消除，并可以得分（规则可以自定义，demo 的规则是消除数量为 n，则得分为 $(n-1)*(n-2)$ ，和之前的游戏并不相同，双五连等情况，交叉点要重复计数）
- 本次移动若得分，则不产生新球，否则会产生三个新球
- 没有任何空位则游戏结束
- 游戏得分不需要记录在文件中，每次开始均从 0 开始即可

菜单项 4: 在 cmd 伪图形界面上画出框架（无分隔线）及初始的五个球

- demo 程序为了看清楚，加了延时，实现时可以不加
- 彩球的颜色、背景色等不需要和 demo 一样
- demo 程序加了设置字体的操作，该部分内容会下发样例程序（不需要自己查资料）

菜单项 5: 在 cmd 伪图形界面上画出框架（有分隔线）及初始的五个球

- 要求同菜单项 4

菜单项 6: 在 cmd 伪图形界面上显示 60%的球，用鼠标选择要移动的球及目的位置，完成一个移动

- 鼠标操作为：左键选择，右键退出本小题
- 鼠标移动过程中，要实时显示当前移动到 $n*n$ 矩阵的哪个位置（行：A-I，列：1-9），放在边框线上不算
- 移动过程需要完整的移动轨迹显示，动画效果必须跨越分隔线
- 如果寻找最佳移动路线不做要求，只要能找到即可（另：demo 中的移动路线很奇葩，但保证了每个位置只移动一次，你作业中找到的路线不能比这个再差了☺）

菜单项 7: 在 cmd 伪图形界面上实现完整版的游戏

- 要求用菜单项 6
- 右侧显示得分、预告彩球以及统计信息

3、提供 90-b2-demo.exe 程序供参考

4、整个程序，**不允许**使用任何形式的全局变量/数组/指针，**允许**使用全局的宏定义或常变量

5、本题需要在 cmd 窗口中做出简单的图形显示（伪图形界面），包括得到当前光标的位置、设置字体与字号、鼠标操作等，这部分内容提供源程序供参考，源程序文件共三个，放在一个项目中编译运行即可，具体说明如下：

cmd_console_tools.cpp: 伪图形界面下基本功能函数的具体实现

cmd_console_tools.h : 伪图形界面下基本功能函数的函数声明

cmd_console_test.cpp : 测试用例

说明：_tools.cpp 和_tools.h 兼容上次汉诺塔给出的同名文件

6、本次作业的要求由以下若干文件组成，具体命名规则如下：

cmd_console_tools.cpp: 同上说明（不准修改，不需提交）

cmd_console_tools.h : 同上说明（不准修改，不需提交）

90-b2-main.cpp : 主函数及菜单部分函数（需要提交）

90-b2-base.cpp : 放内部数组方式实现的各函数（需要提交）

90-b2-console.cpp : 放 cmd 图形界面方式实现的各函数（需要提交）

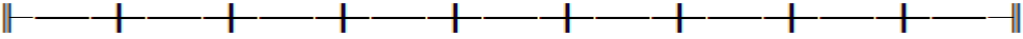
90-b2-tools.cpp : 放一些内部数组/图形方式公用的函数，如判断结束等（需要提交）

90-b2.h : 放上述源程序文件的公用声明部分及其它所需内容（需要提交）

说明：检查时会用各人的上述需提交文件和另两个公共文件一起编译，命名出错则不得分

7、不需要记录最高得分

【特别说明：】


本题中各子题目中要求行/列数可输入，目的并不是降低/增加游戏难度，仅仅是为了让大家在写程序时尽量不要把一些变量值固定（例：循环终值等固定为 9），或者虽然采用 `const int max_row=9` 等方式使固定值方便修改，但在考虑问题是仍然是按 9 的定值去考虑（例如：打印边框线时，采用这种方式把分隔线固定 )，具体请大家自行在作业过程中体会

【函数的分解与使用限制:】

- 1、请参考综合题 1-汉诺塔综合演示的要求，尽量使各菜单项的程序共用函数，用参数解决细节差异
- 2、参数解决差异时，仍然不建议用 if-else/switch-case 等简单方法分解，例如：画 7-9 列的框线时，不能采用下面这种形式，而应该用循环打印整个框线，根据循环值决定框线的长短

```
switch(col) {
```

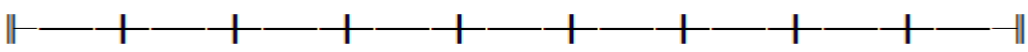
```
    case 7:
```

```
        输出 
```

```
    case 8:
```

```
        ...
```

```
    case 9:
```

```
        输出 
```

```
}
```

- 3、共用函数中，均允许调用其它函数，基本原则就是高效完成程序，减少冗余代码
- 4、**建议：**尽量保证每个函数（包括 main）不要超过 50 行
- 5、**函数分解合理规范的，可以给予最多 2 分的额外加分，本加分项不需要额外提交程序，通过检查源程序后给出相应得分**

【额外加分:】

- 1、“函数的分解与使用限制:”中说明的 2 分
- 2、目前的 cmd_console_tools.cpp 中提供的鼠标操作函数屏蔽了键盘，即鼠标操作时按键是无效的，如果能修改该函数（read_mouse）使能同时读键盘输入（函数名规定为 read_mouse_and_key），并且支持 F3(打开/关闭预告彩球)/F4(重新开始一局)/F5(打开/关闭统计信息)，可以得到 1 分
- 3、目前的鼠标操作，双击操作前会截取到一次单击，左右键同时按下后，会截取到一次左键单击或右键单击，能解决此问题的，可以得到 1 分（提供 demo 程序供参考）

【说明:】第 2、3 项的加分操作，在菜单项的第 8/9 项加入即可，其中第 8 项要求在第 7 项的基础上加入读键，其它不变，第 9 项可以将 cmd_console_test.cpp 中的鼠标部分测试用例移植过来（未完成加分作业的同学，不允许出现菜单项 8/9，否则扣除本题 10% 的分数!!!）

【实验报告:】

本次作业还需要完成对应的实验报告，具体要求另行下发

【作业要求:】

- 1、**2017 年元月 5 日前**网上提交本次作业
- 2、每题所占平时成绩的具体分值见网页
- 3、超过截止时间提交作业会自动扣除相应的分数，具体见网页上的说明