

**【注意:】**

- 1、本次作业**不允许**使用尚未讲授过的任何后续课程的知识点，包括但不限于指针、引用、结构体、类等概念!!!
- 2、已学过的知识中，**不允许**使用 goto，**不允许**使用全局变量，**不允许**使用 C++ 的 string 变量
- 3、**不允许**使用 scanf/printf 进行输入/输出
- 4、要做到“0 errors, 0 warnings”

**综合题 1: 游戏“消灭星星”的实现**

**【参考网址:】** <http://www.7k7k.com/swf/111027.htm> (需要支持 FLASH 的浏览器)

Android/ios 版可自行搜索并下载，不同版本可能规则略有不同

**【玩法说明:】**

- 1、初始在 MxN 的区域内 (M/N 的值要求可设，下面例子中均为 5\*5) 产生随机值，随机值为 1-5 (等概率)，表示不同颜色的星星，然后选中其中一个坐标，依次进行如下步骤的操作：

例如：某次游戏开始后，初值如下所示

	0	1	2	3	4
A	2	1	3	3	4
B	1	3	2	2	1
C	3	4	3	4	5
D	1	1	5	3	2
E	1	2	3	5	2

**【注:】** 列最大为 10，因此 0-9，输入时对应 1 个数字即可

- 2、找到与该坐标相邻的所有相同值 (规则：上下左右值相同，非 0)

例如：

	0	1	2	3	4
A	2	1	3	3	4
B	1	3	2	2	1
C	3	4	3	4	5
D	1	1	5	3	2
E	1	2	3	5	2

输入/选中 D1

	0	1	2	3	4
A	2	1	3	3	4
B	1	3	2	2	1
C	3	4	3	4	5
D	1	1	5	3	2
E	1	2	3	5	2

- 3、将该坐标及相邻的所有相同值均消除 (置 0: 表示该位置无星星)

例如：

	0	1	2	3	4
A	2	1	3	3	4
B	1	3	2	2	1
C	3	4	3	4	5
D	1	1	5	3	2
E	1	2	3	5	2

合并后

	0	1	2	3	4
A	2	1	3	3	4
B	1	3	2	2	1
C	3	4	3	4	5
D	0	0	5	3	2
E	0	2	3	5	2

4、所有列的值跳过 0 自然下落（0 只留在最上面若干行）

例如：

		0	1	2	3	4
A	2	1	3	3	4	
B	1	3	2	2	1	
C	3	4	3	4	5	
D	0	0	5	3	2	
E	0	2	3	5	2	

下落后

		0	1	2	3	4
A	0	0	3	3	4	
B	0	1	2	2	1	
C	2	3	3	4	5	
D	1	4	5	3	2	
E	3	2	3	5	2	

5、如果该列全部为 0，则所有右侧列（可能多列）向左平移，最右侧用 0 填充即可

例如：

		0	1	2	3	4
A	0	0	1	0	0	
B	0	0	2	0	0	
C	2	0	3	0	5	
D	1	0	5	3	2	
E	3	0	3	5	2	

平移后

		0	1	2	3	4
A	0	1	0	0	0	
B	0	2	0	0	0	
C	2	3	0	5	0	
D	1	5	3	2	0	
E	3	3	5	2	0	

6、重复步骤 2-5，直到所有位置均无法找到相邻的相同值（非 0），则本关结束

例如：下图状态则提示本关结束

		0	1	2	3	4
A	0	0	0	0	0	
B	0	0	0	0	0	
C	0	1	0	0	0	
D	0	4	3	0	0	
E	5	2	5	0	0	

7、一关结束后，游戏不要结束，提示后继续下一关，规则相同

### 【游戏规则说明：】

#### 1、判断规则

- 输入/选中的坐标位置处不允许值为 0
- 如果输入坐标位置无相邻相同值（非 0），要提示出错并重新输入
- 如果整个数组无相邻位置值（非 0）相等，提示游戏结束
- 如果一关结束，则游戏不结束，继续下一关

#### 2、分数累加规则

- 本次新增得分 = 消除个数<sup>2</sup>\*5 （也可以自定义规则）
- 一关结束后，根据剩下的星星数（num）给予奖励得分，num≥10 得分为 0，num<10 则奖励得分=(10-num)\*180
- 每关是否设置通关分数（例如：设置 1000 分，如果达到，给出奖励分）可自行决定
- 游戏过程中是否出现宝物、奖励分可自行决定
- 是否需要声音效果可自行决定

### 【子题目划分：】

为降低难度，本题分解为多个小题（菜单形式，大小写不敏感），每小题完成就能取得相应的分数

#### 子题目 A：命令行方式找出可消除项并标识

- 键盘输入行列数，随机产生数组，随后输入要消除的行列坐标，找出所有可消除的位置，并用不同颜色显示出来
- 本小题中在一个矩阵中找相邻位置相同数的方法要求用一个函数实现，该函数的实现为**非递归/递归**形式均可

#### 子题目 B：命令行完成一次消除（分步骤显示）

- 找出所有可消除项后，确定是否需要合并
  - “N”：放弃本次操作，重新输入坐标并继续
  - “Y”：完成本次消除
  - “Q”：放弃游戏
- 如果某次消除并除 0 后，无法找到任何可消除位置，则自动提示游戏结束
- 每次合成操作包括查找相邻项、消除相邻项、计算得分、下落消除 0、平移右侧列，要在程序实现时逐步打印出来

#### 子题目 C：命令行完成一关（分步骤显示）

- 每次合成操作的要求同子题目 2，需要分步骤打印出来
- 完成一关后，给出提示信息（包括剩余的星星数量）

#### 子题目 D：伪图形界面下用鼠标选择一个色块（无分隔线）

- 键盘输入行列数，随机产生数组，将数组的内容在 cmd 窗口中用伪图形显示出来
- 要求 cmd 窗口的大小随着输入的行列数动态变化（高度宽度可适当放大 2-3 行/列）
- 不同数字的前景色/背景色各不相同，具体设置可自行定义
- 画图过程中适当加延时，具体延时值可自行定义，看清楚过程即可（建议比 demo 快）
- 鼠标移动时，经过的色块用**反显**方式（框线用不同颜色显示）以区分，同时下方显示经过的行列值；如果鼠标移出范围，则显示非法位置，同时当前色块的**反显取消**
- 一个色块占用多行多列（demo 中为 3 行 6 列），在同一色块的行列范围内移动时，要求色块**不能闪烁**
- 鼠标在合理位置单击后，显示选择的行列值并结束
- 同时支持键盘操作，用箭头键实现“当前”色块（**反显**方式）的选择，按回车确认
- 键盘操作的初始位置定为左上角，用**反显**方式标注“当前”色块
- 键盘操作情况下越过边界后采用环绕的方式（例如：右箭头向右移动，到达最右侧后，绕回左侧）
- 当鼠标在合理位置时，用键盘要能将“当前”色块移开，而不能反复被鼠标重新定位回去

#### 子题目 E：伪图形界面下用鼠标选择一个色块（有分隔线）

- 鼠标停在色块中间的分割线时，算非法位置
- 其余要求同子题目 D

#### 子题目 F：伪图形界面下完成一次合成（分步骤）

- 箭头键/鼠标移动选择色块，回车键/鼠标单击选定消除位置，并将所有可消除的色块标注出来
- 选定后按回车/再次单击鼠标则进行一次消除，如果需要取消本次选定并重新选择，可以再次用箭头键移动/鼠标移出当前色块范围
- 一次消除操作包括消除相邻项、下落消除 0、右侧列平移、计算得分的操作，要在程序实现时逐步展现出来，其中下落必须有动画效果

### 子题目 G: 伪图形界面完整版

- 每次消除操作的若干步骤间不再需要按键执行，但仍需要有动画效果
- 已被消除的位置，箭头键/鼠标移动算**非法位置**，要跳过
- 如果某次消除后无法找到可消除位置，则提示本关结束
- 一关结束后给出奖励得分，案件后继续下一关
- cmd 窗口的上下各有一个状态栏，显示得分、目标、操作提示等，具体内容可以和 demo 不同，表达清楚即可

#### 【不强制要求的内容:】

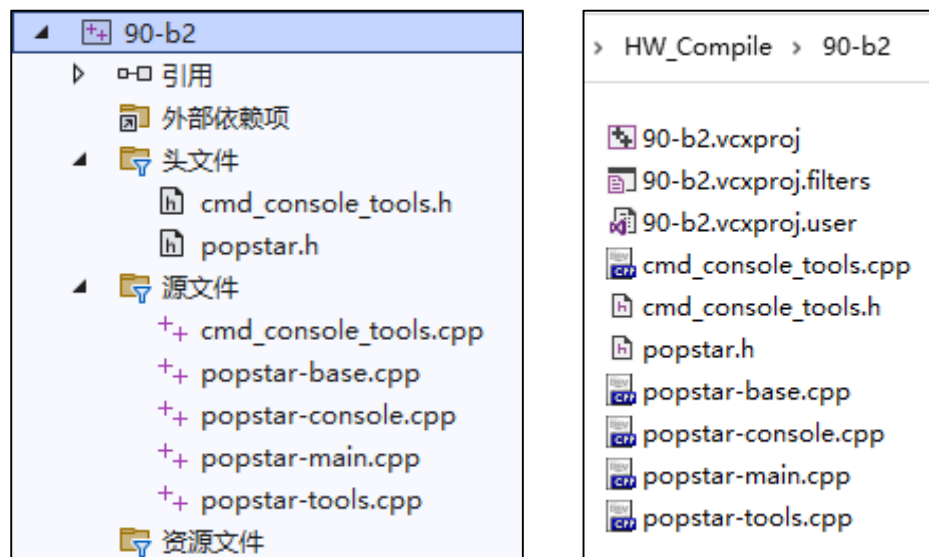
- 1、标记相同值所用的内部数组无强制要求
- 2、字体、字号等无强制要求
- 3、画边框的顺序无强制要求
- 4、延时快慢无强制要求（建议设置比 demo 小），但必须达到动画效果
- 5、下落的顺序（行列哪个优先）、平移的顺序（同列的上下行哪个优先）等无强制要求
- 6、各种提示信息、状态栏的内容等无强制要求
- 7、被标识项、边框、每种数字、选中色块等的颜色无强制要求
- 8、出错时的各种提示无强制要求，清晰明了即可

#### 【实现要求:】

- 1、完成本作业时尽量按程序功能分解为若干函数，注意函数参数的设置，使本作业的多个小题尽量共用一个函数
- 2、完成本作业时应进行综合考虑，使后面的小题尽量能使用前面小题的现有代码（**注：作业开始时，合理划分函数，到后期调试阶段，会极大的提高工作效率并使查错容易，如果单纯为了追求快速做完而不断复制/粘贴同类函数，反而会后期卡住并且无法修改，良好的工程习惯会带来效率的提高**）
- 3、本次大作业的项目命名及提交要求：需要提交四个源程序文件和一个头文件，并严格按照下面的规则命名，具体如下：

popstar-main.cpp : 主函数及菜单部分函数  
popstar-base.cpp : 放内部数组方式实现的各函数  
popstar-console.cpp : 放 cmd 图形界面方式实现的各函数  
popstar-tools.cpp : 放一些内部数组/图形方式公用的函数，如判断结束等  
popstar.h : 放上述源程序文件的公用声明部分及其它所需内容

说明：① 在 VS 中建立一个项目 90-b2，放入 7 个文件，其中 5 个新建，2 个已有，要求编译生成的 exe 文件名**必须是** 90-b2.exe（左图为项目，右图为对应目录）



- ② 伪图形界面工具函数集 `cmd_console_tools.h/.cpp` 仍然使用汉诺塔大作业下发的版本，不允许修改，也不需要提交，检查作业时，会将原始的 `.h/.cpp` 放入后编译，出错则不得分
- ③ 新建的 5 个文件（4 个 `cpp`+1 个 `h`）需要提交，网页上只有一个文件有分数，该分数即本次作业的总分，本题得分按实现功能总体评价而不是按各文件分别给分（例：提交后编译时若 `popstar-base.cpp` 报 `error` 错，则本题总得分为 0 分，而不是仅 `popstar-base.cpp` 为 0 分）
- ④ **五个文件必须全部提交，否则编译错误会导致得分为 0 !!!**
- 4、提供 `90-b2-demo.exe` 供参考（旧版控制台，选项中去掉“快速编辑模式”和“插入模式”）
- 5、整个程序，**不允许**使用任何形式的全局变量/数组/指针，**允许**使用全局宏定义或常变量

#### 【函数提炼的建议（非强制）：】

- 菜单函数
- 查找指定坐标位置及周围相同项的函数
- 判断本关是否结束的函数
- 内部数组进行位置移动的函数，包括从上至下（单色块）、从左至右两种情况（整列左移）
- 命令行下打印数组的函数（包括反显、特殊位置标注等）
- 上下两个状态栏的内容显示函数
- 显示初始外框架的函数
  - 外框架的行数及列数要动态定义
  - 外框架**必须是**中文字符边框，**不能是**英文制表符
  - 外框架的屏幕高度和宽度的计算要动态适应
  - 外框架可选择含分割线/不含分割线的形式
  - 参数解决差异时，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，**不建议**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  ┌───┴───┴───┴───┴───┴───┴───┴───┐
}
```
- 画一个色块的函数
  - 色块的宽度和高度建议能动态定义（通过参数传入），虽然本次游戏的方块是固定的 3 行 6 列（星星是中文★，宽度为 2 列）
  - 色块边框**必须是**中文字符边框，不能是英文制表符
  - 色块的边框同样允许实心/空心的边框线
  - 色块的颜色包括正常/当前/选中等各种形态
- 色块进行位置移动的函数，包括从上至下（单色块）、从左至右两种情况（整列左移）
- 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
- 建议：**尽量保证每个函数（包括 `main`）不要超过 50 行

#### 【编译器要求：】

仅 VS2022 通过即可

### 【控制台要求:】

必须是 Windows 控制台主机的旧版控制台，选项中去掉“快速编辑模式”和“插入模式”

### 【分辨率要求:】

在 1920\*1080 的屏幕下（FHD）显示正常

### 【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

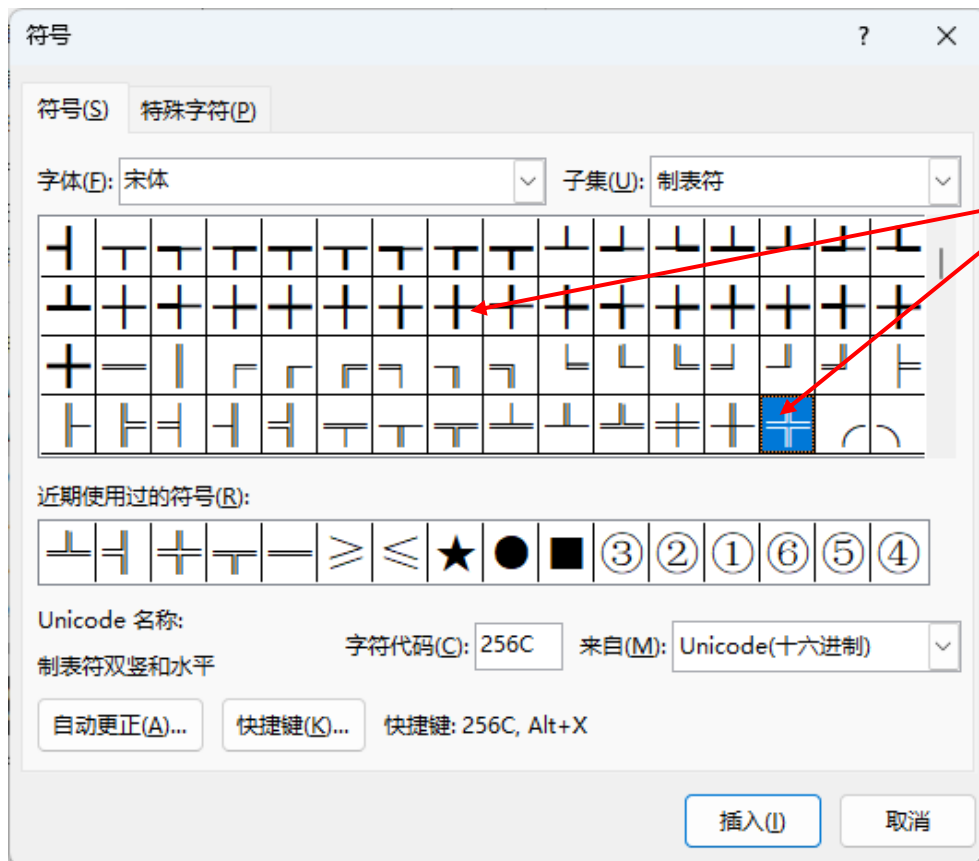
### 【作业要求:】

- 1、6 月 13 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
- 4、大作业期间，每周作业正常下发

### 【附录:】如何画出边框线

- 1、demo 中的框线是中文表格线，每个线段宽度占 2 个字节，可以从 Word 的插入中寻找
- 2、附件“四种线型的中文制表符框架.cpp”已给出四种线型，可任选，无强制要求





自己选  
喜欢的