

## 【注意：】

- 1、本次作业只允许使用到目前为止讲授过的内容（第 6 章及以前）及已完成作业中的补充概念
- 2、已学过的知识中，不允许使用 goto，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、所有输入均需要考虑输入错误的情况，包括同型数据不在指定范围内（例：要求输入[1..12]但输入-2/13 等）以及输入了异型数据（例：需要正整数但输入字符）的情况
- 5、作业必须符合相应的缩进格式，格式分占 10%

## 综合题 2：游戏“合成十”的实现

## 【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，初始随机值为 1-3，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

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

- 2、找到与该坐标相邻的所有相同值（规则：上下左右值相同）

例如：

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

输入 D2

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

- 3、将所有的相邻值合并到指定位置，该位置值+1，被合并后的其余位置置 0

例如：

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

合并后

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

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

下落后

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

5、所有 0 位置随机产生新值（新值的范围为 1-N，其中 N 为当前最大值）

例如：

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

产生新值后

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

6、重复步骤 2-5，直到某个位置的值达到完成标准（例如：10）

7、达到完成标准后，游戏不要结束，提示后继续向更高标准合并，每多达到一个数字，就给出一次提示（例如：11、12、…）

8、如果所有数字都无法合并，则游戏失败

例如：下图状态则提示游戏失败

	1	2	3	4	5
A	2	1	3	1	2
B	1	3	2	4	1
C	3	1	5	1	3
D	1	4	3	7	2
E	5	2	5	6	5

### 【游戏规则说明：】

1、随机值的产生规则

条件	随机数值范围	分布概率
1、初始 2、合并后最大值为 3	1-3	等概率
3、合并后最大值为 4	1-4	1~3：各 30% 4：10%
4、合并后最大值为 5	1-5	1~3：各 25% 4：15% 5：10%

5、合并后最大值为 6	1-6	1~4: 各 20% 5 : 15% 6 : 5%
6、合并后最大值为 x (x>6)	1-x	1~x-3: 80%/(x-3) 均分后取整 x-2 : 10% x-1 : 5% x : 5%

## 2、判断规则

- 如果输入坐标位置无相邻相同数，要提示出错并重新输入
- 如果整个数组无相邻位置值相等，提示游戏结束
- 如果达到合成目标，则提示完成，游戏不结束，继续向更高目标前进

## 3、分数累加规则

本次新增得分 = 消除值\*消除个数\*3

例如：

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

合并后

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

本次新增得分 = 3 \* 5 \* 3 = 45

## 【实现要求：】

- 1、完成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，使多个小题尽量共用一个函数
- 2、完成作业时进行综合考虑，使后面的小题尽量能使用前面小题的现有代码
- 3、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 **不兼容** 上次汉诺塔给出的同名文件（本次的**版本修订**将 hout/hin 做为**静态全局变量**放在了\_tools.cpp 中，外部调用不再带参数）

- 4、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**

- 5、提供 90-b2.exe 程序供参考

- 6、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

7、为了降低难度，本题分解为多个小题（菜单形式），每小题完成后就能够取得相应的分数

子题目 1：命令行方式找出可合并项并标识

- 键盘输入行列数，随机产生数组，随后输入要合并的行列坐标，找出所有可合并的位置，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**形式
- 提示：如何标记相邻数据，可以用另一个相同大小的二维数组来记录标记（demo 中输出了查找结果数组，供参考）

子题目 2：命令行方式找出可合并项并标识

- 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**递归**形式，其余同子题目 1

子题目 3：命令行方式下完成一次合成

- 找出所有可合并位置后，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”：完成本次合并
  - “Q”：放弃游戏
- 如果某次合并后无法找到可合并位置，则自动提示游戏结束
- 每次合成操作包括查找相邻项、合并相邻项、计算得分、下落消除 0、在 0 位置产生新数据，要在程序实现时逐步打印出来

子题目 4：命令行方式完整版

- 每次合成操作的要求同子题目 3，需要分步骤打印出来
- 合成到预期目标值后，给出提示信息，但不结束，可继续进行游戏

子题目 5：cmd 伪图形窗口显示内部数组的内容（数字色块间无分隔线）

- 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

子题目 6：cmd 伪图形窗口显示内部数组的内容（数字色块间有分隔线）

- 其余要求同子题目 5

子题目 7：cmd 伪图形窗口下的“当前选择”色块的选择

- 在子题目 6 的基础上，用箭头键实现“当前选择”色块的选择，按回车确认
- 初始位置定为左上角，用不同颜色标注“当前选择”色块
- 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）
- 为方便调试，延时可以取消

子题目 8：cmd 伪图形窗口下完成一次合成

- 箭头键选择色块，回车键选定合成位置，并将所有可合成的色块标注出来（注意合成位置的色块颜色与其他可合成项要有区别）
- 选定后再次按箭头键则取消本次选定，重新选择，按回车则进行一次合成
- 一次合成操作包括合并相邻项、下落消除 0、0 位置产生新数据、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
- 下落的顺序（行列哪个优先）、填充的顺序（行列哪个优先）可以与 demo 不同

### 子题目 9: cmd 伪图形窗口完整版

- 每次合成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
- 达到合成目标后游戏不结束，合成目标+1 后游戏继续
- 如果某次合并后无法找到可合并位置，则提示游戏结束
- 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

#### 【无强制要求的内容:】

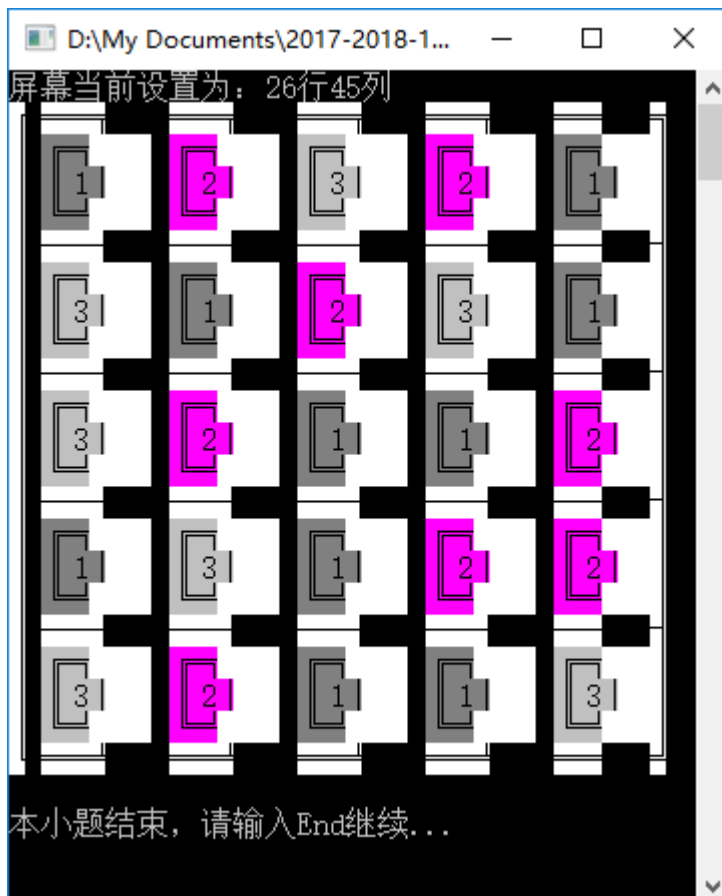
- 1、 标记相同值所用的内部数组无强制要求
- 2、 字体、字号等无强制要求
- 3、 画边框的顺序无强制要求
- 4、 延时快慢无强制要求（建议设置比 demo 小），但必须达到动画效果
- 5、 下落消除 0 的行列顺序、0 位置产生新数据的行列顺序等无强制要求
- 6、 各种提示信息、状态栏的内容等无强制要求
- 7、 被标识项、边框、每种数字、选中色块等的颜色无强制要求
- 8、 出错时的各种提示无强制要求，清晰明了即可

#### 【作业要求:】

- 1、 仅需要在VS2017下编译通过即可，要做到“0 errors, 0 warnings”
- 2、 **2018 年元月 7 日前**网上提交本次作业
- 3、 每题所占平时成绩的具体分值见网页
- 4、 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

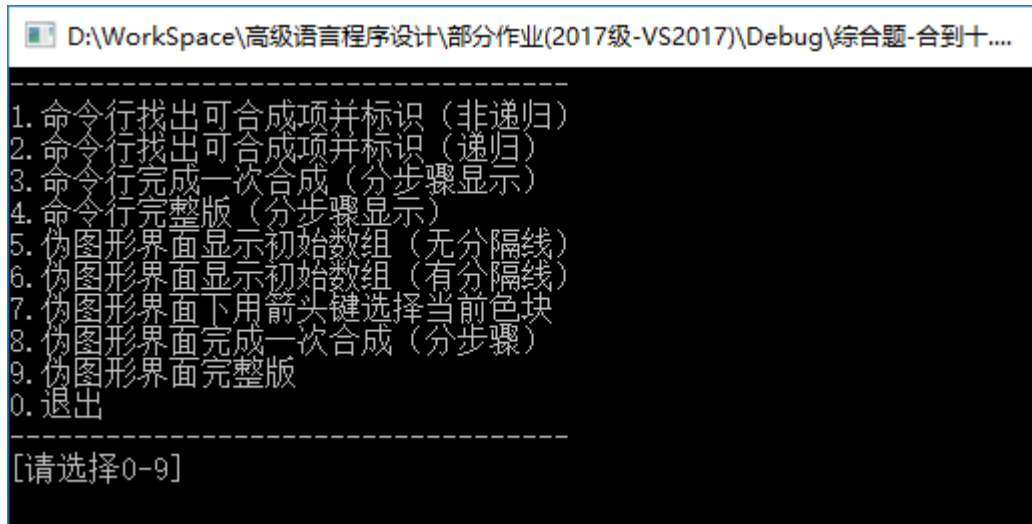
#### 【附录A: 使Windwos 10的cmd窗口能正常显示demo程序中的边框（制表符）】

- 1、 目前demo程序的边框（制表符）显示为乱码

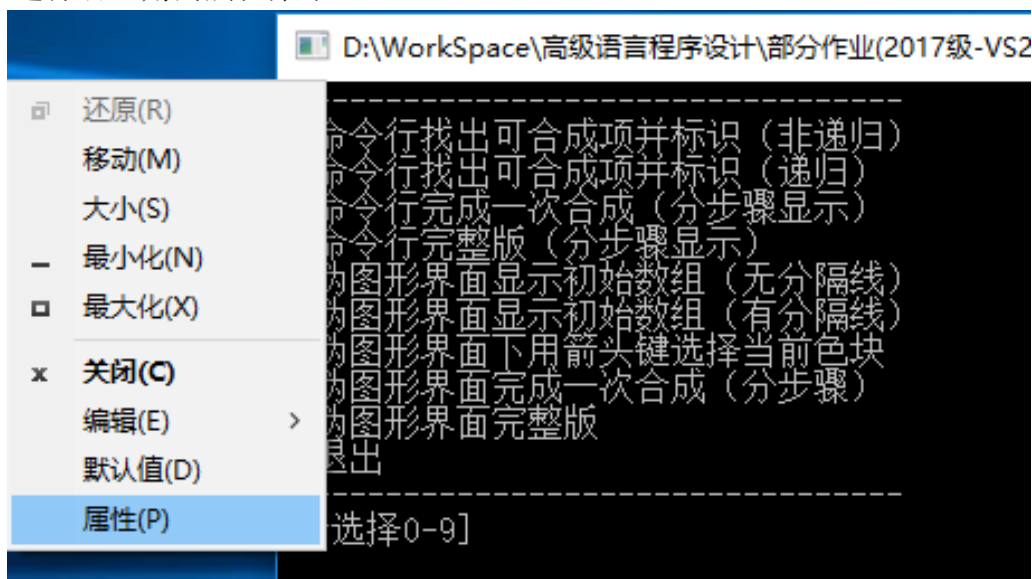


注：本截图为第6小题显示

## 2、启动cmd窗口



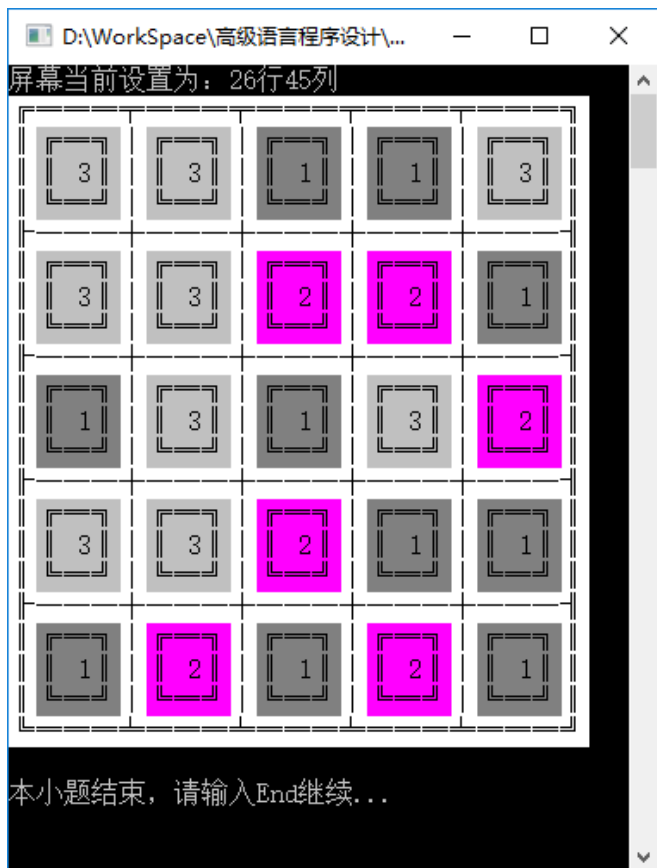
## 3、选择右上角的属性菜单



## 4、按下面的红色框的要求进行设置，按“设置”



5、关闭cmd窗口!!!，再次运行，demo程序的边框显示正常



注：本截图为第6小题显示

6、自己的程序调试时，也照此设置即可

### 【附录 B:】如何在设置屏幕宽度和高度的命令中使用参数

引入: `system("mode con cols=100 lines=40")` 可以改变屏幕的高度和宽度，  
如果 `100/40` 是变量该如何处理？

解决方法:

```
int row, col;  
char cmd[80];
```

通过键盘输入或计算等方式给 `col/row` 赋相应的整数值

```
sprintf(cmd, "mode con cols=%d lines=%d", col, row);  
cout << cmd << endl; //这句话只是为了查看 cmd 数组的内容，不需要时可以删除  
system(cmd); //系统调用生成的 cmd 字符串
```

注：1、`cmd_console_tools` 函数中的 `setconsoleborder` 已经采用了这种方法

2、其他生成字符串时需要参数的也可用此方法（例：色块中的数字 `2/10` 显示宽度不等）

## 【附录 C:】边框线的画法

demo 中的框线是中文表格线，每个线段占 2 个字节，可以从 Word 的插入中寻找

