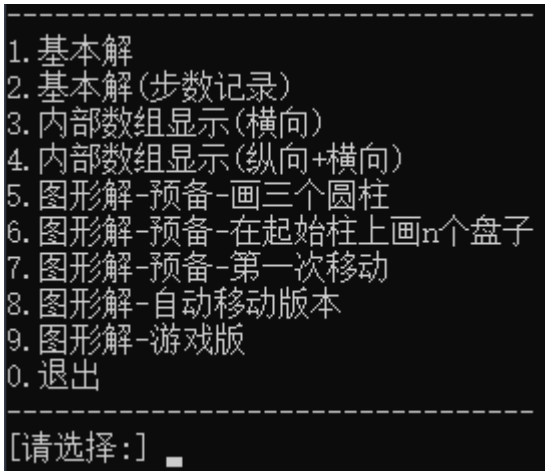
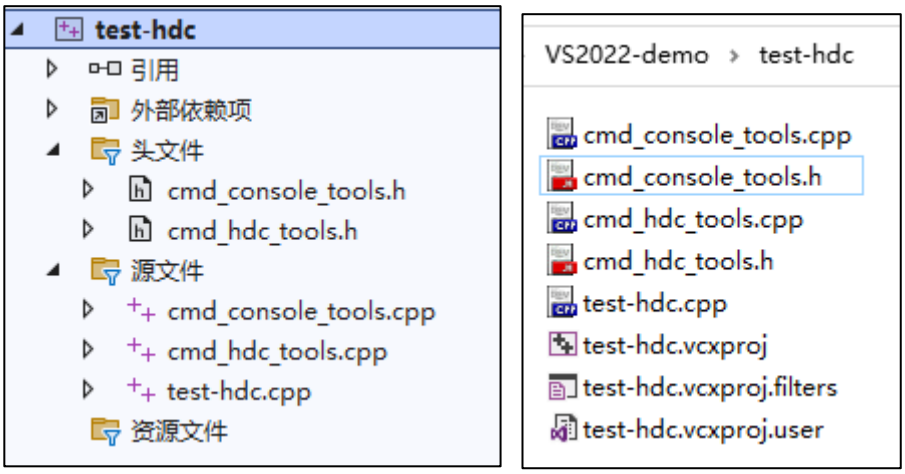
【注意:】
1、 本次作业不允许使用尚未讲授过的任何后续课程的知识点，包括但不限于指针、引用、结构体、类等概念!!!
2、 已学过的知识中，不允许使用 goto，不允许使用全局变量，不允许使用 C++的 string 变量
3、 不允许使用 scanf/printf 进行输入/输出
4、 要做到“0 errors, 0 warnings”

综合题 1：汉诺塔综合演示
【要求:】1、将之前做的所有汉诺塔的各小题集成在一个程序中，用菜单方式进行选择，并加入图形化演示的要求（cmd 窗口中简单的图形显示，后续均称为伪图形界面）



2、提供 90-b1-demo.exe 供参考（有正常版、胖版、瘦版共三个版本，差异见后）
3、伪图形界面中字符工具函数集的学习：附件中有 2 个文件，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
注：用于清屏及移动字符光标，5-b7 作业用过
4、伪图形界面中图形工具函数集的学习：附件中有 3 个文件，说明如下
cmd_hdc_tools.cpp ： 伪图形界面下基本功能函数的具体实现
cmd_hdc_tools.h ： 伪图形界面下基本功能函数的函数声明
test-hdc.cpp ： 测试用例
说明：① 在 VS 中建立一个项目 test-hdc，将这 5 个文件放入，即可编译并运行测试用例，每个函数的具体功能及使用方法请阅读源程序及测试用例



注：要求五个文件放在同一项目中（左）并且在同一目录下（右）

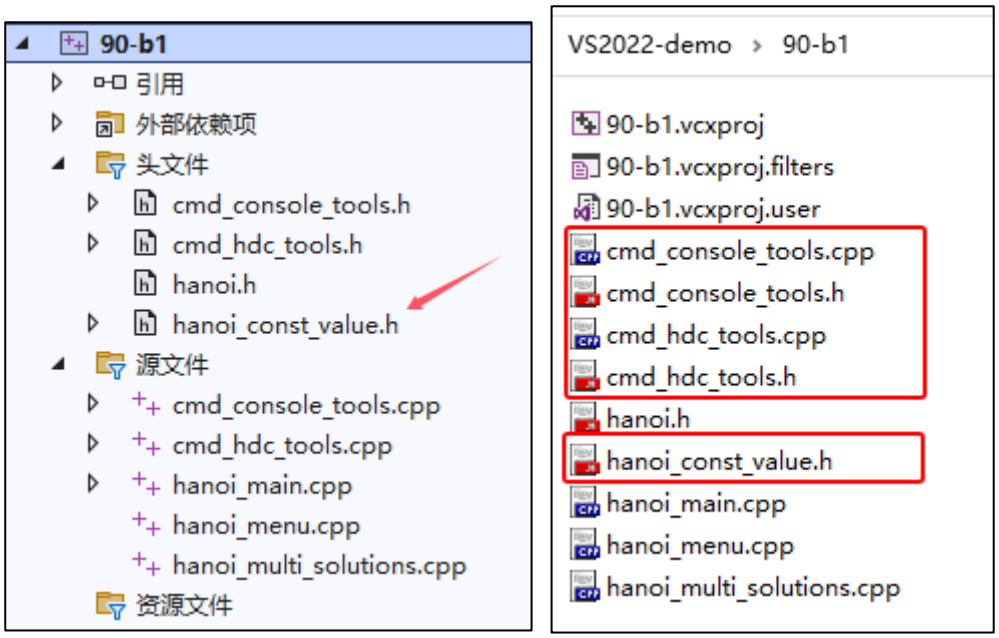
② cmd_console_tools 中的 cct_* /cmd_hdc_tools 中的 hdc_*系列函数已经能满足本次作业的所有需求，**不需要**再学习并额外添加伪图形界面类的函数

③ 如果阅读时**源代码与注释有不一致**的地方，以可编译的**源码为准**

4、本次大作业的项目命名及提交要求： 附件中有 9 个文件，说明如下

cmd_console_tools.cpp	：同上
cmd_console_tools.h	：同上
cmd_hdc_tools.cpp	：同上
cmd_hdc_tools.h	：同上
hanoi.h	：本项目头文件
hanoi_const_value.h	：本项目头文件（只读常量定义，具体见后）
hanoi_menu.cpp	：菜单的显示与选择
hanoi_multiple_solutions.cpp	：菜单中各项汉诺塔演示的实现
hanoi_main.cpp	：main 函数

说明：① 在 VS 中建立一个项目 90-b1，将这 9 个文件放入（下发文档中的文件名要去掉前缀，文件名不要修改），要求编译生成的 exe 文件名**必须是** 90-b1.exe



② **要求 9 个文件放在同一项目中（上图左）并且在同一目录下（上图右），否则可能会编译出错导致得分为 0 !!!**

③ 上图右红框中的五个文件不允许修改，也不需要提交，检查作业时，会将原始的.h/.cpp 放入后编译，**出错则得分为 0 !!!**

③ 其余 4 个文件的功能要求及限制请具体查看每个文件，这 4 个文件需要提交，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 hanoi.h 报 error 错，则本题总得分为 0 分，而不是仅 hanoi.h 为 0 分）

④ **四个文件必须全部提交，否则编译错误会导致得分为 0 !!!**

5、下列内容**允许**使用全局变量记录，其余均不允许（全局 const 变量/#define 宏定义的数量不受限制，任意使用）

- 总移动步数 ：1 个全局简单变量/静态局部变量
- 圆柱上现有圆盘的编号：3 个全局一维数组或 1 个全局二维数组
- 圆柱上现有圆盘的数量：3 个全局简单变量或 1 个全局一维数组
- 延时 ：1 个全局简单变量

6、为了降低难度，伪图形界面部分拆分为若干小题（菜单项 5-9），完成每个小题能够取得相应的分数

菜单项 1-4：之前小作业的整合，会占一定的分数（具体待定，max=30%）

菜单项 5：在屏幕上画出三根圆柱

- 为方便观察实现过程，需要加延时

菜单项 6：假设三根圆柱的编号从左到右分别为 ABC，要求输入起始圆柱的编号（A-C），圆盘的数量（限制在 1-10 之间），在起始圆柱上从小到大画出 n 个圆盘，每个圆盘的颜色各不相同

- 为方便观察实现过程，需要加延时

菜单项 7：在菜单项 6 的基础上，完成第一个圆盘的移动

- 第一次移动并不一定是从源柱->目标柱，也可能是源柱->中间柱
- 移动的时候，有些延时是必须加的，否则无法模拟出移动效果，具体的可以自行在实现过程中体会
- 不允许直接在两个圆柱间移动，必须先上移、再平移、再下移（具体参考 demo）

菜单项 8：汉诺塔演示过程的完整实现

- 每次圆盘的移动方式也必须是上移、平移、下移

菜单项 9：汉诺塔游戏（人工操作移动步骤）

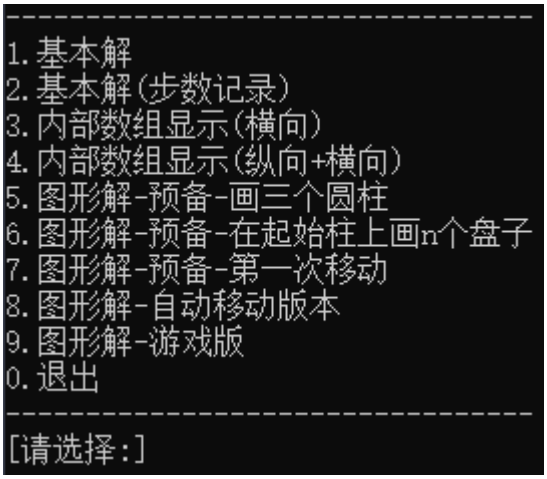
- 每次键盘输入两个字母(A-C)之间，大小写均可，表示本次移动的源柱和目标柱
- 移动时要检查合理性，若不符合移动规则（大盘压小盘、源柱为空等）要提示出错并重输，每次合理的移动都必须记录步数
- 每次圆盘的移动方式也必须是上移、平移、下移
- 待所有盘子按序移动到结束柱则提示“游戏结束”
- 本小题不需要调用递归函数

7、屏幕显示要求

- 为方便观察实现过程，需要加延时，延时的系统函数为 Sleep(单位：毫秒)，需要包含头文件<Windows. h>，例如 Sleep(100)表示延时 0.1 秒（demo 的 0 为不延时，1-200 表示延时 1-200ms）
- 输入完成后，用 cct_cls()/hdc_cls()可以清除屏幕上现有的内容，但是该命令只能使用一次，不允许每次移动一个元素就清屏并全部重新输出（直观感受就是屏幕会闪烁），而是只能擦除原有位置，在新位置上输出（例：当前一步操作为 3 从 B 移动到 C，则只能在 B 位置擦除 3，C 位置显示 3，图形方式要求能呈现出动画效果，具体见后面的描述）

【函数的分解与使用限制：】

为了更好地掌握函数的分解与应用技巧，对 hanoi_multiple_solutions.cpp 中的函数的定义和使用做出限制，具体要求见下：



1、 整个程序只允许使用一个递归函数，即菜单项 1/2/3/4/8 必须共用一个递归函数，用参数解决各菜单项不同要求之间的差异，递归函数按一句一行计算（包含独立成行的左右大括号），**不得超过 15 行**

【提示：】横向、纵向数组打印、色块移动等可以通过在递归函数中调用其它函数来实现

2、 菜单项 1/2/3/4/6/7/8 中的输入多个参数必须共用一个函数，用参数解决输入不同内容的问题**（本函数允许使用第 6 章的知识：函数形参为实参的指针，可以同时改变多个实参值）**，菜单项 9 各人看具体情况决定是否共用（即建议共用，但如果分开处理也可以）

3、 菜单项 3/4/8 中的横向输出必须共用一个函数，用参数解决输出位置等差异

4、 菜单项 4/8 中的纵向输出必须共用一个函数，用参数解决输出位置等差异

5、 菜单项 5/6/7/8/9 中画三个柱子的必须共用一个函数

6、 菜单项 7/8/9 中盘子的移动必须共用一个函数

7、 以上的共用函数中，均允许调用其它函数，希望大家在作业过程中体会如何划分函数才能高效完成程序，减少冗余代码

8、 其中 1-4 项需要的函数，不受之前作业的限制（即函数的参数个数、类型可以与之前不同）

9、 **建议：**尽量保证每个函数（包括 main）不要超过 50 行

【无强制要求的内容：】

1、 各种提示信息、状态栏的内容等无强制要求

2、 横向、纵向数组打印时的空格数量、冒号中英文等无强制要求（对齐即可）

3、 出错时的各种提示无强制要求，清晰明了即可

4、 本题是**人工判题**，不是自动判题（即：不必太在意细节处理）

【通过 const 参数的设置来达到可变参数的目的（强制要求）：】

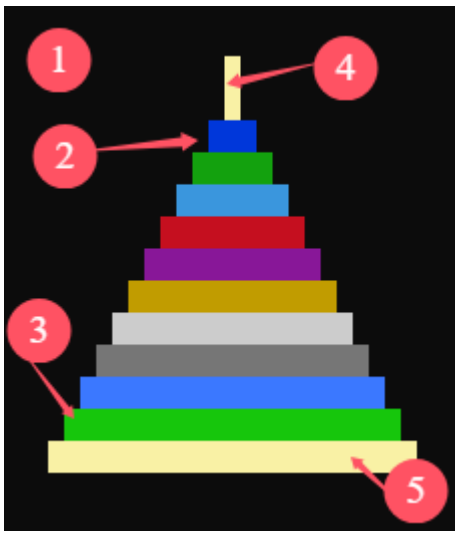
1、 为了能在显示位置、颜色、图形的粗细等方面做到灵活变化（不要在程序中写死!!!），附件自带三个 hanoi_const_value.h，要求将任意一个改名为 hanoi_const_value.h 后，均能编译通过，并且达到对应 demo 的显示效果

hanoi_const_value_胖版.h
hanoi_const_value_瘦版.h
hanoi_const_value_正常.h

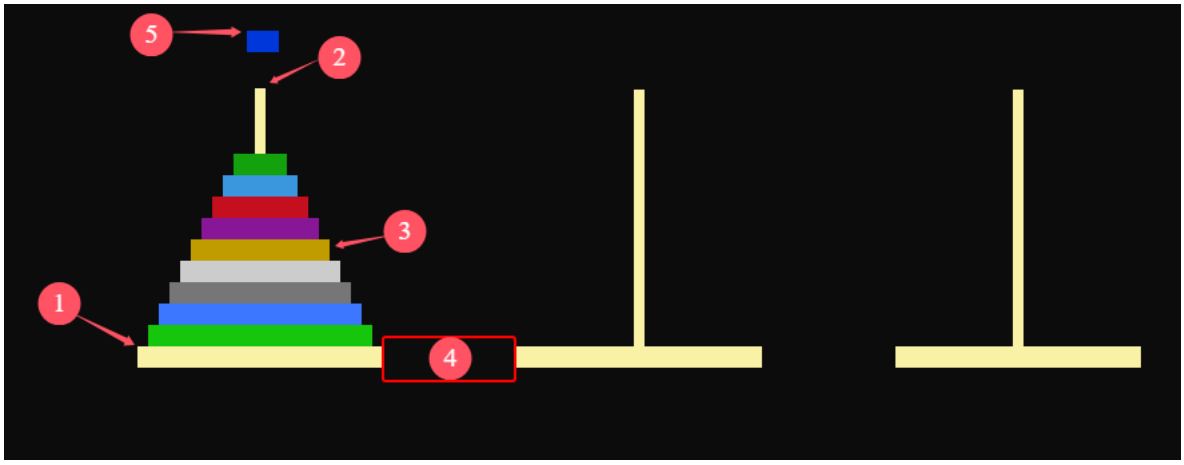
2、 检查作业时，会以新的 hanoi_const_value.h 代入后编译并评分

3、 以 hanoi_const_value_正常.h 为例，说明每个参数的具体含义

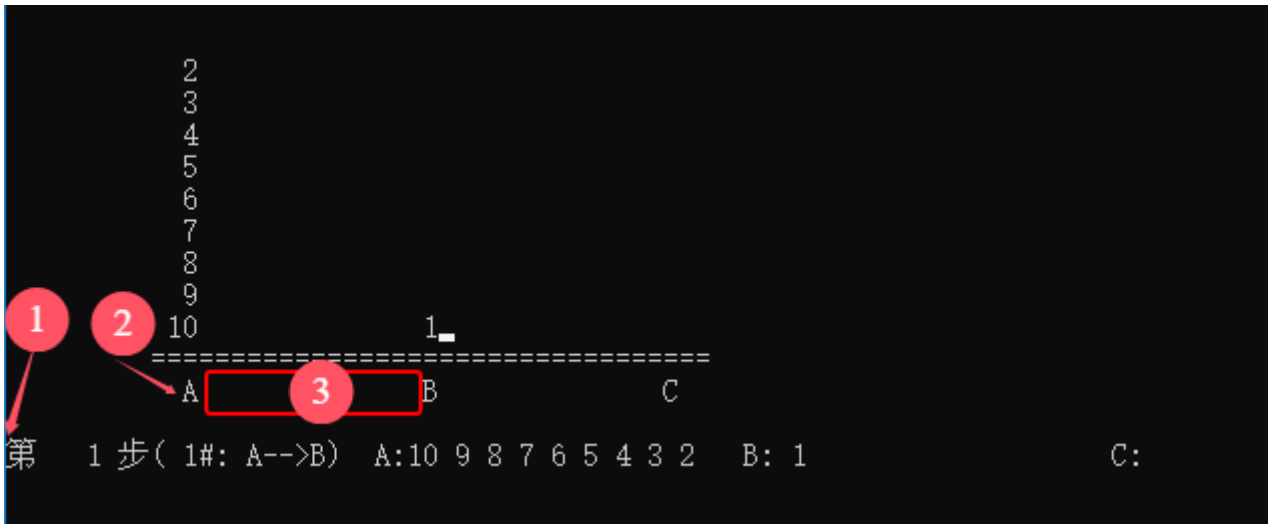
```
/* 定义底座和盘子的颜色*/
const int HDC_COLOR[MAX_LAYER + 2] = {
    RGB(12,12,12),   //①背景底色
    RGB(0,55,218),   //②1#盘(最小的盘子)的颜色
    RGB(19,161,14),
    RGB(58,150,221),
    RGB(197,15,31),
    RGB(136,23,152),
    RGB(193,156,0),
    RGB(204,204,204),
    RGB(118,118,118),
    RGB(59,120,255),
    RGB(22,198,12),   //③10#盘(最大的盘子)的颜色
    RGB(249,241,165) //④⑤基座颜色（底盘+立柱）
}; //0 是底色，1-10 是 10 个盘子的颜色，11 是基座的颜色
```




```
const int HDC_Init_Delay = 1000;//菜单 5/6/7/8 画底座和盘子之间的演示，具体看 demo
const int HDC_Start_X = 100; //① A 柱左上角的 X 坐标（单位：像素点）
const int HDC_Start_Y = 256; //① A 柱左上角的 Y 坐标（单位：像素点）
const int HDC_Base_Width = 8;//② 立柱的宽度（单位：像素点），假设为 w
                             //   1#~10#盘的宽度分别是 3w~21w，底座宽度 23w
const int HDC_Base_High = 16;//③ 盘/底座的高度（单位：像素点），假设为 h，立柱高度为 12h
const int HDC_Top_Y = 20;     //⑤ 盘子移动到最顶上时，上沿的 y 坐标（单位：像素点）
const int HDC_Underpan_Distance = 100; //④ 立柱之间的距离（单位：像素点）
const int HDC_Step_X = 1; //某个盘左/右移动的步进距离（单位：像素点）
const int HDC_Step_Y = 1; //某个盘上/下移动的步进距离（单位：像素点）
                          //通过 demo 的菜单 8，选单步执行方式来观察
```



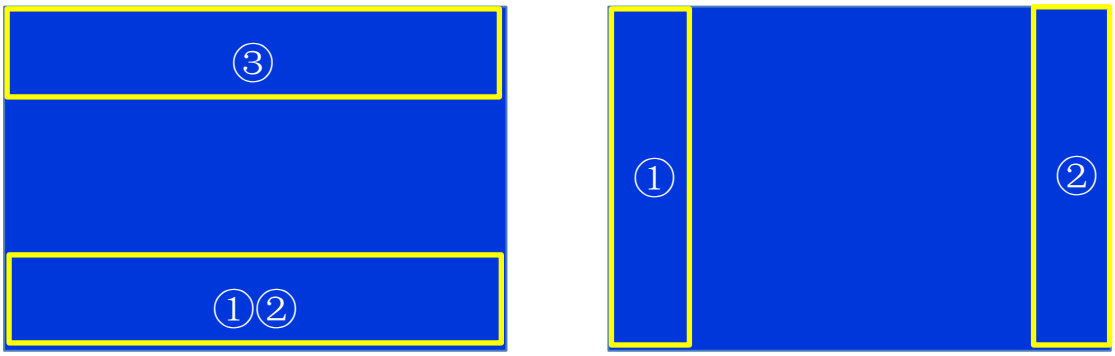
```
const int MenuItem4_Start_X = 0; //①菜单 4 横向数组第一个字符的 X 坐标
const int MenuItem4_Start_Y = 17;//①菜单 4 横向数组第一个字符的 Y 坐标
                                 //单位：字符，一个汉字占两个字符
const int MenuItem8_Start_X = 0; //菜单 8，同上
const int MenuItem8_Start_Y = 34;//菜单 8，同上
const int MenuItem9_Start_X = MenuItem8_Start_X; //菜单 9，同上
const int MenuItem9_Start_Y = MenuItem8_Start_Y; //菜单 9，同上
const int Underpan_A_X_OFFSET = 11; //② 立柱字母 A 相对于横向数组的 X 偏移量（单位：字符）
                                    //   例：11 表示在“第”的右边 11 个字节打印 A
const int Underpan_A_Y_OFFSET = - 2; //② 立柱字母 A 相对于横向数组的 Y 偏移量（单位：字符）
                                     //   例：-2 表示在“第”的上面两行打印 A
const int Underpan_Distance = 15; //③ 立柱 A/B/C 之间的距离（单位：字符，15 表示 14 个空格）
```



【盘子移动的动画效果实现说明：】

以 HDC_Base_Width=4、HDC_Base_High=4、HDC_Top_Y=20、HDC_Step_X=1、HDC_Step_Y=2 为例

1#盘：宽度为 12（3w），高度为 4（1h）



向上移动：① 将下部 2 行（HDC_Step_Y=2）像素点用底色重画（抹除）

② 将下部 2 行（HDC_Step_Y=2）像素点的中间用基座底色重画（恢复立柱）

③ 将上部 2 行（HDC_Step_Y=2）像素点用该色块的颜色重画（显示上移效果）

④ 加 1ms 延时，使 1-3 的重复呈现出动画效果

⑤ 持续上移到 HDC_Top_Y 位置，上移结束

向右移动：① 将左侧 1 列（HDC_Step_X=1）像素点用底色重画（抹除）

② 将上部 1 列（HDC_Step_X=1）像素点用该色块的颜色重画（显示右移效果）

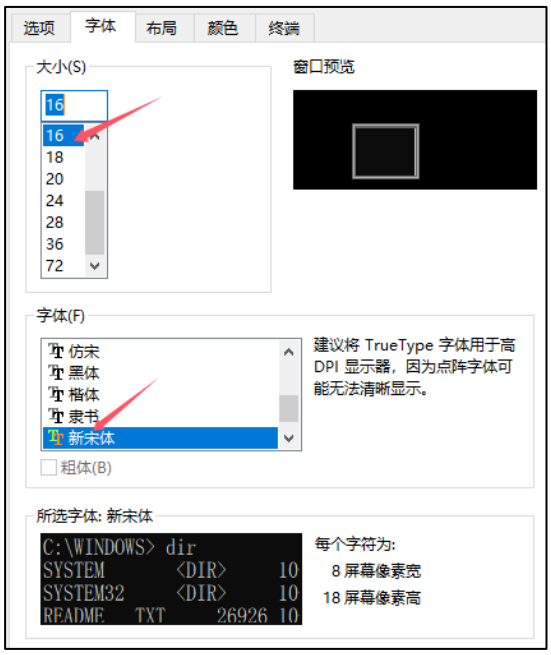
③ 加 1ms 延时，使 1-2 的重复呈现出动画效果

④ 重复右移到本次移动的 dst 对应的立柱中间，右移结束

向下/向左：不再重复

【编译器要求：】
仅 VS2022 通过即可

【控制台要求：】
1、必须是 Windows 控制台主机的新版控制台，字体为“新宋体 16 点阵”，窗口大小“120*40”以上



2、如何将 cmd 窗口由“powershell”改为“Windows 控制台主机”，参见之前的文档/视频
3、如何在新版/旧版控制台之间切换，参见之前的文档/视频

【分辨率要求：】
在 1920*1080 的屏幕下（FHD）显示正常，如果你的笔记本是高分屏（超过 FHD）但是使用了缩放倍率，完成后最好设成分辨率 1920x1080/缩放 100% 验证一下，否则可能影响得分

【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、5 月 25 日前网上提交本次作业（第 13 周周日，两周半时间）
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