

【注意:】

- 1、本次作业**不允许**使用后续课程的知识点, 包括但不限于指针、引用、结构体、类等概念!!!
- 2、除明确要求外, 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 string
- 3、cstdio 及 cmath 中的系统函数**可以**直接使用, 包括课上未介绍过的, 具体可自行查阅相关资料
- 4、除明确要求外, 所有 cpp 源程序不允许使用 scanf/printf 进行输入/输出
- 5、多编译器下均要做到 “0 errors, 0 warnings”
- 6、部分题目要求 C 和 C++两种方式实现, 具体见网页要求
- 7、输出为浮点数且未指定格式的, 均要求为 double 型, C++为 cout 缺省输出, C 为 %lf 的缺省输出
- 8、认真阅读格式要求及扣分说明!!!

【输出格式要求:】

- 1、为方便机器自动判断正确性, 作业有一定的输入输出格式要求 (但不同于竞赛的无任何提示)
- 2、每个题目见具体说明, 必须按要求输入和输出, 不允许有偏差
- 3、没有特别说明的情况下, 最后一行有效输出的最后有一个 endl
- 4、**本次作业的比对要求为 txt_compare 在 --trim right 下与 demo 做到完全一致**

补充:

- 7、题目同 4-b12、5-b6 (汉诺塔), 要求以纵向方式给出完整的移动过程

【要求:】 1、假设圆盘最大数量为 10, 其余输入格式要求同 5-b6

- 2、输入移动延时及是否显示内部数组两项

- 内部数组对应的是 5-b6-1 的格式, 即三个简单变量+三个一维数组方式
- 移动延时, 允许采用一个 int 型静态全局变量
- 延时用系统函数 Sleep 实现 (单位: 毫秒), 需要包含头文件 <Windows. h>, 例如 Sleep(100) 表示延时 0.1 秒
- 是否显示内部数组, **允许**采用一个 int/bool 型静态全局变量

- 3、5-b7 要求三个文件组成

- 5-b7-cct.cpp : 屏幕及光标设置的工具函数
- 5-b7.h : 放 5-b7-cct.cpp 的函数声明, 供 5-b7-main.cpp 调用
- 5-b7-main.cpp: 主函数及实现部分
- 将附件给出 5-b7.h 和 5-b7-cct.cpp 文件放入 Project 中的方法
 - ◆ 在 VS 下先建立 5-b7 项目
 - ◆ 附件给出 5-b7.h 和 5-b7-cct.cpp 文件改名 (**去前缀**) 后放入对应目录中
 - ◆ VS 中选中该项目的“源文件”/“头文件”, 右键菜单选“添加”-“现有项”, 加入已存在的 5-b7-cct.cpp 和 5-b7.h
- 作业只需要提交 5-b7-main.cpp, 其余两个文件不准修改、不需要提交

- 4、其余要求同 5-b6

- 5、给出 5-b7-demo.exe 供参考 (双击执行, 因为带光标移动, 不支持输出重定向, 为了在最后暂停, 允许 main 函数 return 0 前加一个 system("pause");

- 具体的延时时间不要求与 demo 完全相同, 能体现出各档的差异即可
- **人工判题, 细节不必完全按照 demo, 只要能表现出三柱间的移动效果即可**

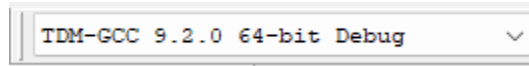
- 6、本题仅要求 VS 实现即可

8、在完成“第 02 模块作业-基础知识题(Part3-理解 IEEE754).pptx”时，对尾数部分的 2 进制小数形式转十进制小数，提供了一个工具“工具：将二进制小数转换为十进制.exe”，现在我们要把这个工具自行实现

【要求：】1、IEEE754 有多个版本，对不同长度的浮点数进行了定义，本题选择三种，具体见下：

类型	字节	bit 位	符号位	指数位	小数位
float	4	32	1	8	23
double	8	64	1	11	52
long double	16	128	1	15	112

【注：】只有 DevC++ 的 64-bit 方式下 long double 才占 16 字节，VS 仅 8 字节



- 2、本题涉及高精度的浮点数计算，因此不能用任何浮点型，只能用数组模拟
- 3、程序首先输出 $2^{-1} \sim 2^{-112}$ 对应的十进制小数形式，随后再输入一个二进制纯小数，输出其十进制小数形式
- 4、给出 5-b8.cpp 基准程序供参考（注：基准程序中给了足够的提示，如果不愿意用，也可以不受基准程序的限制而自行完成，txt_compare 一致即可）
- 5、输出 $2^{-1} \sim 2^{-112}$ 对应的十进制小数形式时，不能是打表形式，只能在给出的 2^{-1} 为 0.5 的基础上循环计算得到 $2^{-2} \sim 2^{-112}$ （打表则期末总分-20）
- 6、输入二进制纯小数的格式要求：必须用. 开头，后面跟 0/1，有任何输入错误则全部重新输入，如果长度超过 112 位则只取前 112 位
- 7、提供 5-b8-demo.exe 供参考（cmd 下运行）

9、给定一个 9*9 的矩阵，判断是否满足数独的解

【要求：】1、矩阵的 81 个数据从键盘输入（可重定向）

- 2、满足数独解的条件是每行/每列/每个小九宫格都只有 1-9 的数字且不重复
- 3、输入错误的处理要求：
 - 每个数字输入完成后，要判断是否正确（1-9 之间），不正确则重输
 - 处理原则：cin 正确但范围不合理，不清缓冲区直接重读；cin 错误则清缓冲区
- 4、给出一个示例数据文件 sudoku.txt，可自行编辑后用于重定向（重定向文件是否允许输入错误？自行思考）
- 5、给出 5-b9-demo.exe 供参考

输出格式要求：多行

Line1: 输入提示

Line2-x: 输入的 9*9=81 个数字

Line x+1: 判断结果“是数独的解”
“不是数独的解”

Microsoft Visual Studio 调试控制台

请输入 9*9 的矩阵，值为 1-9 之间

5	1	6	2	7	4	3	9	8
7	9	3	5	6	8	4	1	2
8	2	4	3	9	1	7	6	5
4	5	1	6	3	7	2	8	9
3	7	2	1	8	9	6	5	4
9	6	8	4	5	2	1	3	7
2	3	5	8	4	6	9	7	1
6	4	9	7	1	5	8	2	3
1	8	7	9	2	3	5	4	6

是数独的解

D:\Workspace\高级语言程序设计\部分作业
按任意键关闭此窗口...

10、 输入年份，打印该年的年历

- 【要求:】
- 1、年份限定在 1900-2100 之间（不考虑输入错误）
 - 2、要求每行输出三个月的月历，而且输出时**只能**按照从上到下、从左到右依次输出，**不允许使用**光标定位函数(cct_gotoxy 或类似函数/退格键等)来重定位光标的位置
 - 3、每个月的月历，最短为 4 行，最长为 6 行，要保证每三个月的月历格式不乱
 - 4、如果输出窗口的高度不足以放下四个季度的月历，需要改变输出窗口的宽度和高度，具体方法见附件
 - 5、**不允许使用**三维（含）以上的数组（其实二维也没有必要，一维足够了）
 - 6、给出示例程序 5-b10-demo.exe 供参考（有延时/非延时两个版本，延时版本在每个日期间加了 Sleep(50)，目的只是为了演示输出顺序，实现非延时版本即可）
- 【提示:】**本题的难点**在于不使用光标定位函数时如何同时计算并记录多个月份的输出位置

Microsoft Visual Studio 调试控制台

请输入年份[1900-2100]
2020
2020年的日历: 英文冒号 月份居中

1月

2月

3月

4月

5月

6月

7月

8月

9月

10月

11月

12月

日期输出为宽度4，左对齐，因此最后一列数字的最后2~3个空格

每个季度之间空一行

日期与星期左对齐，所有间隔均为空格填充，不存在 tab 等字符

空两行（程序在最后多空两行即可，不要考虑系统自己加的空行）

【编译器要求:】

		编译器VS	编译器Dev
5-b7-main.cpp	汉诺塔-每步详细(横向+纵向)-仅提交此文	Y	/
5-b8.cpp	高精度幂计算	Y	Y
5-b9.cpp	数独判断	Y	Y
5-b10.cpp	打印年历	Y	Y

【作业要求:】

- 1、5月9日前网上提交本次作业（因5.1小长假影响，本次作业时长为两周）
- 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

【附件:】Windows控制台主机方式下，如何调整cmd窗口的大小

