

§ 3. 程序设计初步 - 输入输出及顺序结构部分

【注意：】

- 1、本次作业只能定义简单变量并用顺序结构实现
- 2、不允许使用尚未讲授过的任何后续课程中的知识点，包括但不限于比较、逻辑、条件运算符以及分支、循环、函数、数组、结构体、类等相关概念!!!
- 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

书：

- P. 80 习题 2（题目前后有矛盾，不需要从键盘输入，直接用给定值即可）
P. 80 习题 3

补充：

- 1、用 C 语言的 `printf` 函数完成与 C++ 的 `cout` 一样的输出（不需要输出中文提示部分）
说明：① 附件中的 `3-bl.cpp` 已包含了完整的 `cout` 输出，不准改动，只准改动首行的个人信息以及 `printf` 函数所在行，使其与上面 `cout` 的输出要求完全一致即可
② 按回车键依次执行，直到全部结束
③ C 语言补充资料中无 `printf` 对整数带符号位 (+) 的输出方法，需自行查找相关资料
（不允许用 `if-else` 之类的分支语句输出符号位，否则本题得分直接为 0）
- 2、键盘输入一个 $[1..30000]$ 之间的整数 (假设输入保证正确，不必考虑各种输入错误的情况)，要求把每一位数字分解并打印出来

例如：输入 12345，则输出为：

万位：1
千位：2
百位：3
十位：4
个位：5

例如：输入 307，则输出为：

万位：0
千位：0
百位：3
十位：0
个位：7

- 3、键盘输入一个 $[0 \dots 100 \text{ 亿})$ 之间的浮点数，要求把每一位数字分解并打印出来

例如：输入 8307238123.07，则输出为：

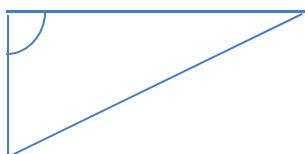
十亿位：8
亿位：3
千万位：0
百万位：7
十万位：2
万位：3
千位：8
百位：1
十位：2
个位：3
十分位：0
百分位：7

- 提示：① 浮点数需要考虑误差 **（特别提示：此处是难度之所在）**
② 输入用 `cin >> 某类型变量` 的形式一次性输入，**不允许** 采用整数/小数分开输入等方法
③ 不必考虑输入超范围及错误的情况，输入时小数点后最多两位即可

- ④ 该数的整数部分最大值可能超过 int/long 的最大可表示范围，**不允许**使用 64 位及以上的整数，**不允许**采用把该数**整体扩大**若干倍的方法
- ⑤ 要求下面的测试数据（共 24 个）全部通过

| | | | |
|---------------|---------------|---------------|--------------|
| 999999999.99 | 999999999.90 | 999999999.09 | 9900000000 |
| 8912003005.78 | 2501200350.03 | 1203056740.00 | 203056740.20 |
| 23000056.82 | 3051200.72 | 301000.35 | 10001.34 |
| 8070.23 | 9876.54 | 803.03 | 12.30 |
| 10.03 | 9.30 | 7.03 | 0.35 |
| 0.30 | 0.07 | 0.03 | 0 |

- 4、键盘依次输入三角形的两边及夹角，给出三角形的面积，边长为正整数，夹角为正整数，表示角度，输出结果保留到小数点后 3 位



例：输入 3 4 90，输出 6
输入 2 2 60，输出 1.732

【用本次作业体验 C 程序和 C++程序的区别：】

- 除 3-b1 外，所有程序均要求两种格式完成：*.c 和*.cpp
- *.c 的是纯 C 语言程序，不能包含任何 C++的内容；*.cpp 是 C++程序，允许包含 C 及 C++的内容

下面是纯 C 格式和 C++格式的“helloworld”程序示例

| | |
|--|---|
| <pre>#include <stdio.h> int main() { printf("hello,world\n"); return 0; }</pre> | <pre>#include <iostream> using namespace std; int main() { cout << "hello,world" << endl; return 0; }</pre> |
|--|---|

- 在 VS2017、CodeBlocks、Dev C++中，如果源程序名为*.c，则自动采用纯 C 编译器；如果源程序名为*.cpp，则会自动采用 C++编译器
- 在 Linux 中，用 gcc -o 编译 C 程序；用 c++ -o ***编译 C++程序

【作业要求：】

- 本次作业全部要求为 4 编译器同时通过（0 error, 0 warning）
- 10 月 25 日前**网上提交本次作业
- 每题所占平时成绩的具体分值见网页
- 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