

基础练习：错误处理

1. (错误, 书中练习 P. 98, 2-6,9) 下面的程序是获得摄氏温度值并将其转化为绝对温度。但这些代码有很多错误, 找出这些错误, 指出并修改它们。

```
double ctok(double c)    // converts Celsius to Kelvin
{
    int k = c + 273.15;
    return int
}

int main()
{
    double c = 0;    // declare input variable
    cin >> d;        // retrieve temperature to input variable
    double k = ctok("c");    // convert temperature
    Cout << k << endl;    // print out temperature
}
```

- 绝对零度是能够达到的最低温度, 即 -273.15°C 或者 0K 。即使上面的程序是正确的, 当输入一个低于这个值的温度时, 程序也应输出错误结果。检查一下, 当输入一个低于 -273.15°C 的数值时, 主程序是否产生错误。
 - 重做练习, 但这次把错误处理放在 `ctok()` 中。
 - 给这个程序增加一些功能, 使它也可以把绝对温度转化为摄氏温度。
 - 编写一个程序, 它可以实现绝对温度转化为华氏温度和华氏温度转换为绝对温度 (公式见 4.3.3 节)。用估计的方法 (参见 5.8 节) 看看你的结果是否合理。如果结果不能表现为整型的话, 输出一个错误信息。
2. (错误, 书中练习 P. 98, 7) 一个一元二次方程的形式如下:

$$a \cdot x^2 + b \cdot x + c = 0$$

解这个方程, 用到二次公式:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

这里面有一个问题: 如果 $b^2 - 4ac$ 小于零的话, 它将出错。编写一个可以解一元二次方程的程序。建立一个可以计算二次方程根的函数。给定 a 、 b 、 c , 如果 $b^2 - 4ac$ 小于 0 就抛出一个异常。让程序的主函数调用这个函数, 如果有错误由主函数捕获异常。当程序发现方程没有实根的时候, 输出相应的信息。你如何确定程序的结果是合理的? 你能检验结果的正确性吗?

3. **（错误，书中练习 P.98, 11）** 编写程序，输出尽量长的斐波那契序列，也就是说，序列的开始是 1 1 2 3 5 8 13 21 34，下一个数是前两个数之和。找出 int 能允许的最大的斐波那契数。
4. **（错误，书中练习 P. 98, 14）** 从标准输入中读入（星期，数值）对，例如：
星期二 23 星期五 56 星期二 -3 星期四 99
将一个星期中每一天对应的数值存入一个 `vector<int>` 中。输出 7 个向量中的值。打印每个 `vector` 中的值的总和。忽略输入中的非法日期，例如 `Funday`，但是接受同义词，例如 `Mon` 和 `Monday`。输出被拒绝数值的个数。