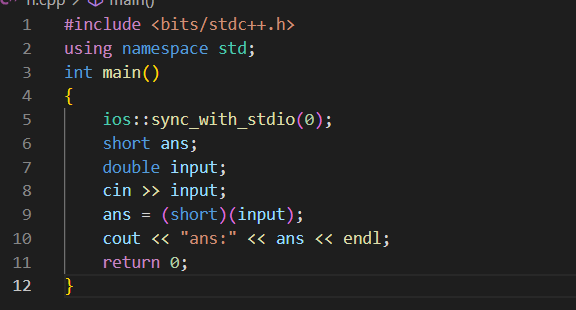
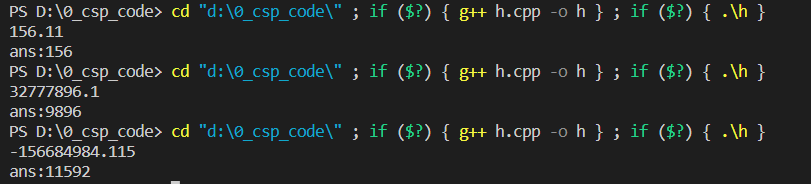
**①复现代码问题：**

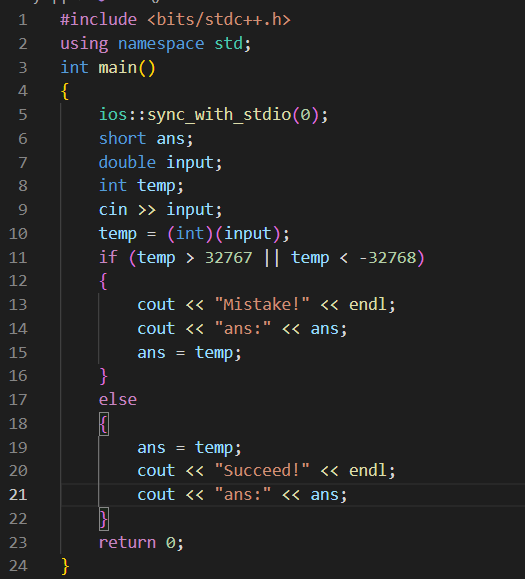
**代码：**



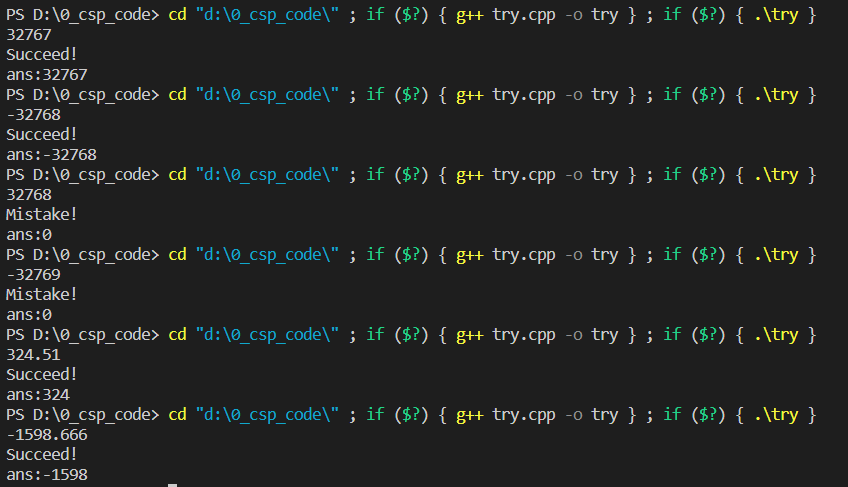
**结果：**



**更改后的代码：**



**更改后的结果：**



**②所谓的“bug”被发现的难易度：**

如果是一个人进行代码的书写的话，可能难以发现。因为问题比较小，只要用数据测试的时候才会知道有BUG，但是具体在哪还是要查很久。

如果是两个人或多人进行代码的书写的话，很难查找出来这个BUG，因为BUG虽然很小，而且一眼就能看出来，但是没有人会怀疑是数据出现了误差，更多的会认为是逻辑之间出现了问题，而不会注意这个问题。而且是每个人负责自己的部分，很难知道是哪部分出现了问题，导致更难知道是这个“bug”引发的错误。

**③团队中的人能发现它吗：**

很难发现，但是也有一定的概率可以发现并纠正。我认为可能需要由2个人或3个人可以系统性地检查代码，来找到这个BUG。因为这个BUG属于不常见的BUG，大概率程序运行时也不会有异常报出，更多的是可能出现错误的结果。所以要发现它最好是系统性地检查，因为可能导致后续也有一些出现错误。

**④避免或降低此类问题的风险的措施：**

可以有多种解决方案。

其一，在书写代码的时候，不要照抄原先的代码，全部手写。但是这样的工程量很大，不建议使用。

其二，在复制之前的代码的时候，考虑一下两个工程之间的差异，根据差异，多尝试几组数据验证是否出现不一致，来降低错误的概率。但是这样也有一定的概率会出现错误，且难以排查。

其三，在所有的代码的工作完成之后，由不同的人负责不同的模块部分，进行检查和排查。有专人负责检查对应数据转换及数据传送之间的问题，这样可以一定程度地避免或降低这类风险出现的概率。