**实验1 表达式翻译器**

**一、方案转化**

表达式文法及其翻译方案定义如下：

*expr* → *expr + term* {*print*(‘+’)}

| *expr - term* {*print*(‘-’)}

| *term*

*term* → *term* **\*** *factor* {*print*(‘\*’)}

*| term* **/** *factor* {*print*(‘/’)}

*| factor*

*factor* →**(**  *expr* **)**

**| id** *{ print(lexeme*) }

**| num** { *print*(*tokenval*) }

消除左递归之后的翻译方案如下：

*expr* → *term rest1*

*rest1* → + term {*print*(‘+’)}  *rest1*

*| - term* {*print*(‘-’)} *rest1*

*|*

*term* → *factor rest2*

*rest2* → \* factor {*print*(‘\*’)}  *rest2*

*| / factor* {*print*(‘/’)} *rest2*

*|*

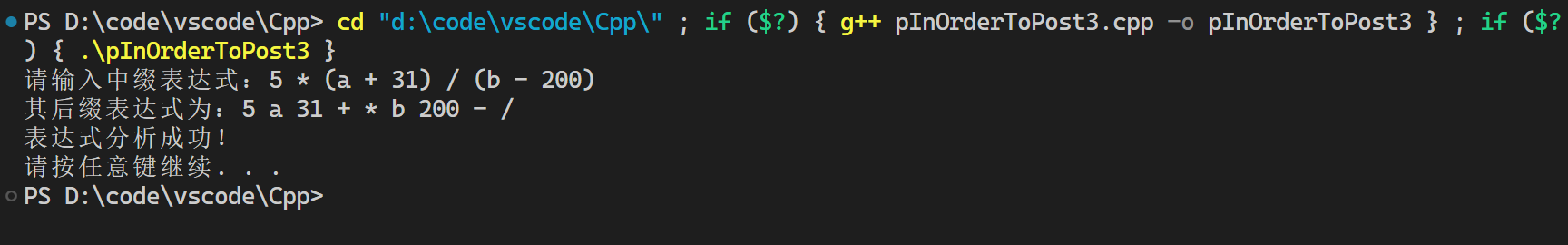
*factor* →**(**  *expr* **)**

**| id** *{ print*(*lexeme*) }

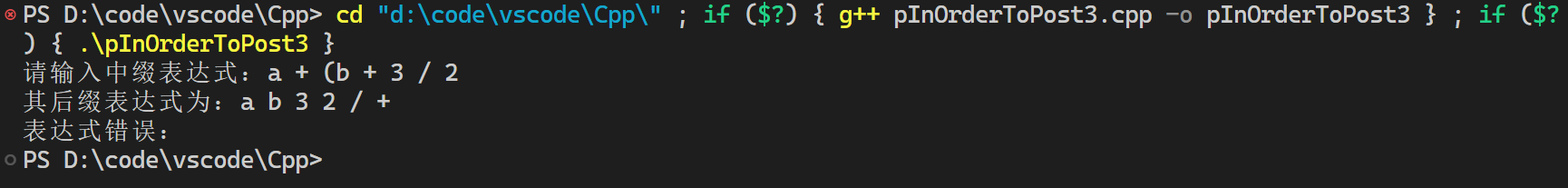
**| num** { *print*(*tokenval*) }

**二、实验结果**

合法输入



不合法输入



**三、实验问题及解决方法**

问题：

在编写代码的过程中，由于Rest1函数和Rest2函数及结构及内容相似，考虑将两个函数合为1个函数，但实际尝试后发现运行结果错误。

解决方法：

若将两个函数合并，则会出现expr推出乘除法，term推出加减法的情况，不符合翻译方案，会导致错误结果，所以应该分为两个函数来写。