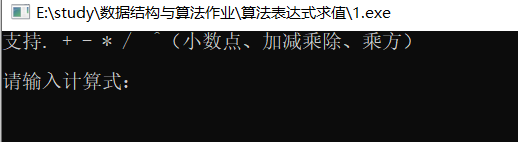
**Project2 实验报告**

20337263 俞泽斌

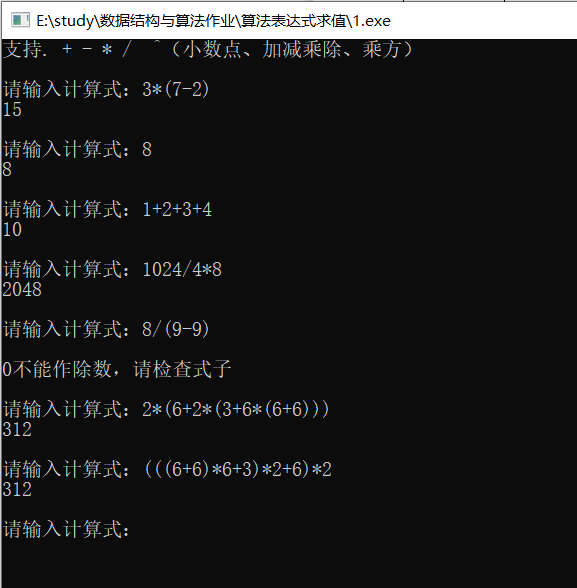
1. 程序功能简要说明。

本程序主要是通过栈的方式，将算术表达式通过栈的存取，压栈、入栈等操作，先是将普通表达式转化成逆波兰表达式，所以也可以支持逆波兰表达式的输出，后将逆波兰表达式通过计算优先级等进行判断并输出运算结果。

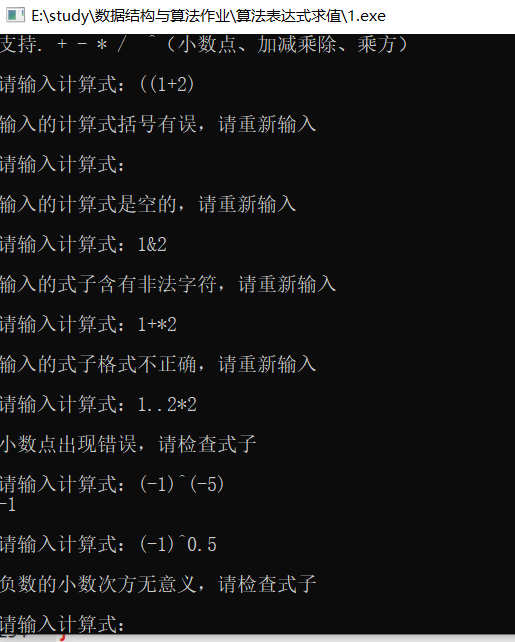
1. 程序运行截图，包括计算功能演示、部分实际运行结果展示、命令行或交互式界面效果等。

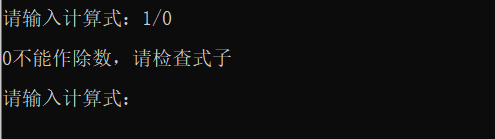


这里是本程序的开始界面，采用的是比较简单的命令行的模式，简要说明了本程序支持的运算和方法，就是输入要计算的计算式。



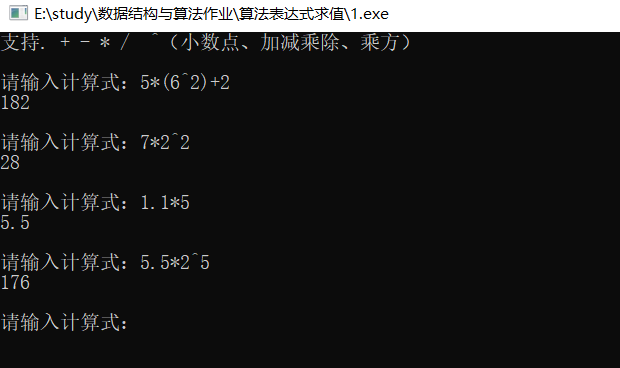
这里是关于老师文档里给的几个测试数据的输入和输出，可以看到在加减乘除方面是没有问题的，也在除数为0的时候抛出了异常，并进行输出提醒操作。其实在其他异常情况下也对异常进行抛出和处理操作，截图如下：





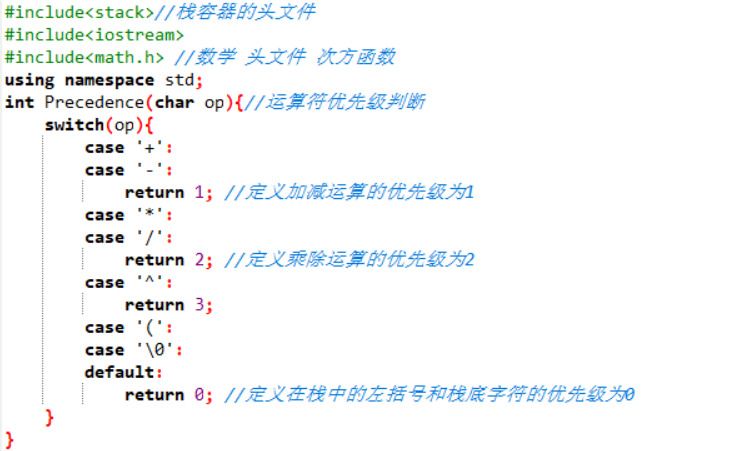
这里是关于各个异常的处理，分为7类，分别是1、括号的不匹配就是两个左括号一个右括号，无法计算，2、计算式子为空，3、计算式子含有非法字符，也就是在开始界面里没有出现的不支持的运算，4、计算式子的格式不正确，主要是将两个计算符连起来了，5、小数点不正确，主要是输入了两个小数点连在一起和开始就是小数点即小数点前没有数的异常情况，6、0不能做除数，这也是计算方面的错误了，7、负数的小数次方，也就是相当于对负数开平方，这当然也是没有意义的。这里就都是对于计算方面的异常的处理。

以下是关于乘方和小数点等老师文本里未给出测试数据的测试。

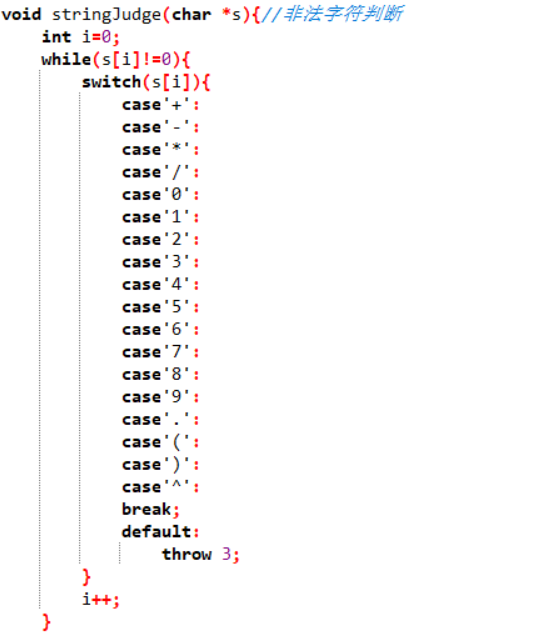


主要也就是对于乘方和小数点的数据进行操作了，然后加上了对于乘方和小数点合起来使用的测试数据，表现一切正常。

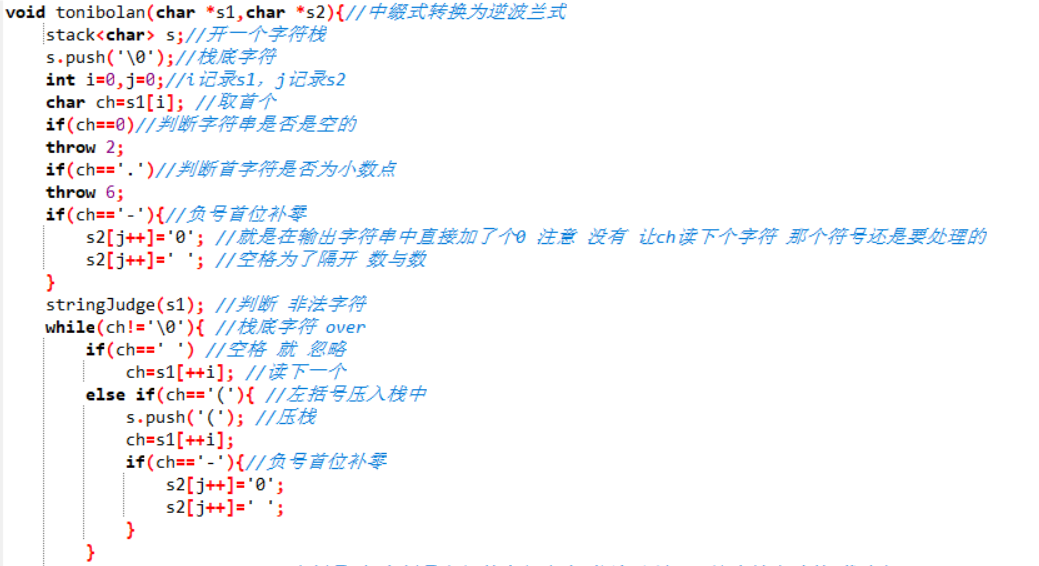
1. 部分关键代码及其说明。

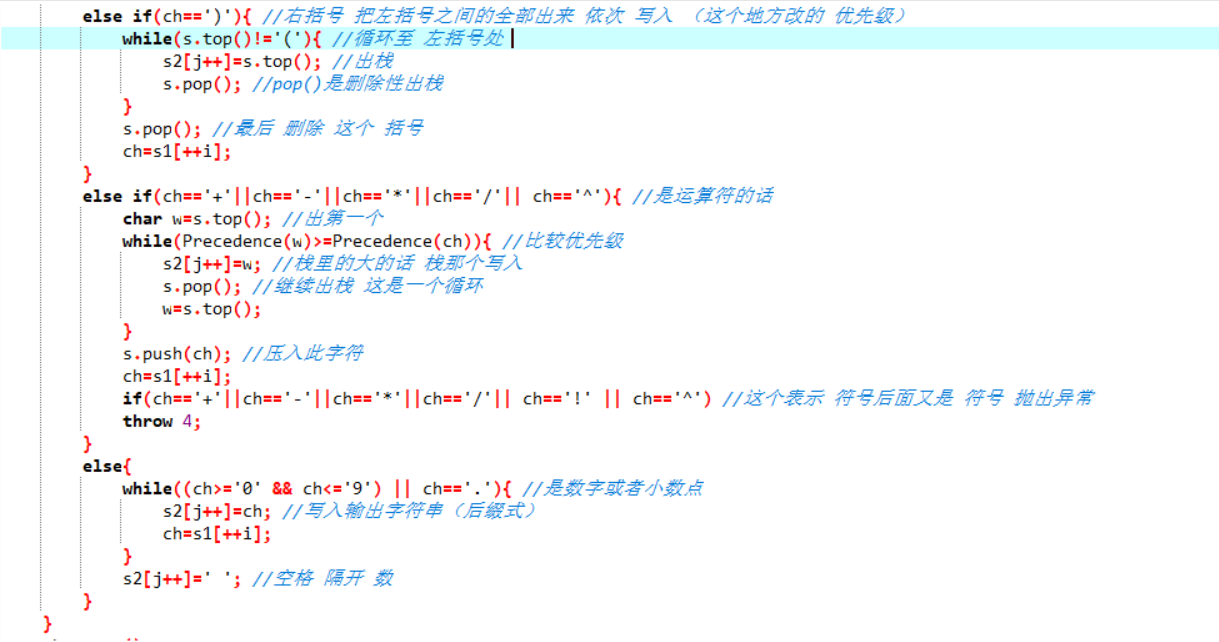


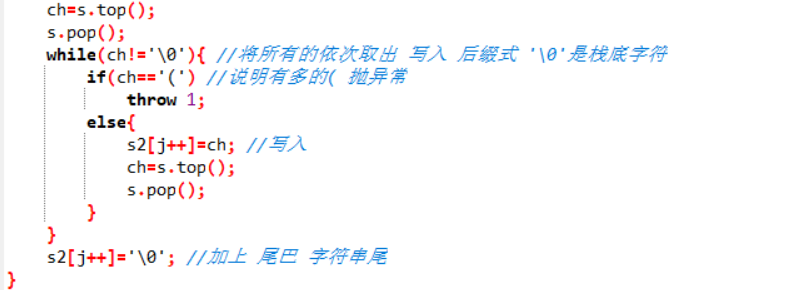
这里是关于头文件以及运算符优先级的判断，其实二维数组的表太复杂，对于栈内和外有区别，这里就简单的对于其他进行定义。



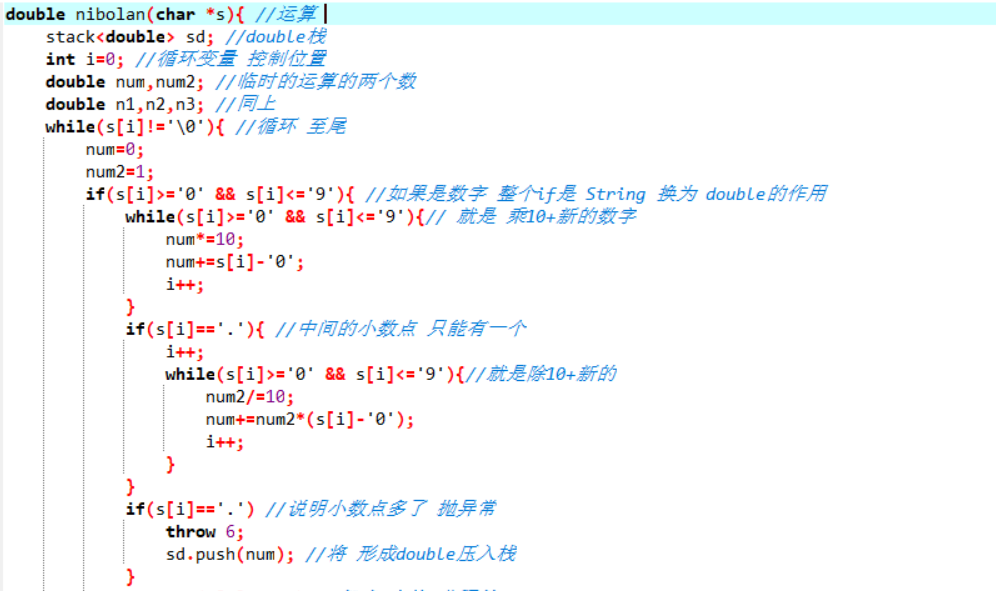
这里是关于非法字符的判断，也就是抛出的第一个异常点。

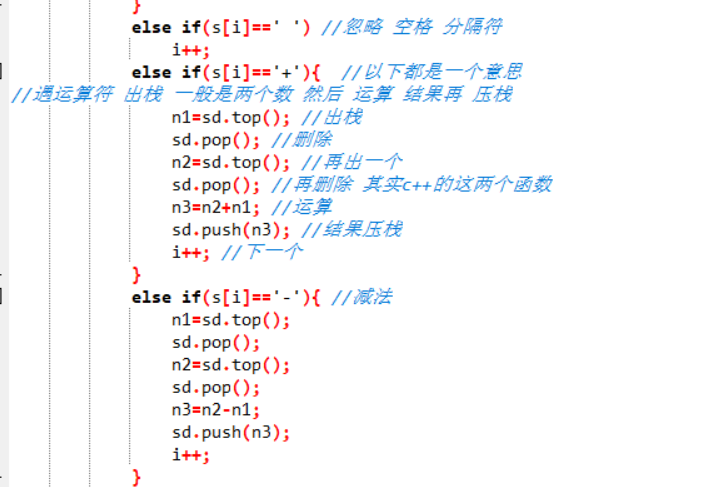


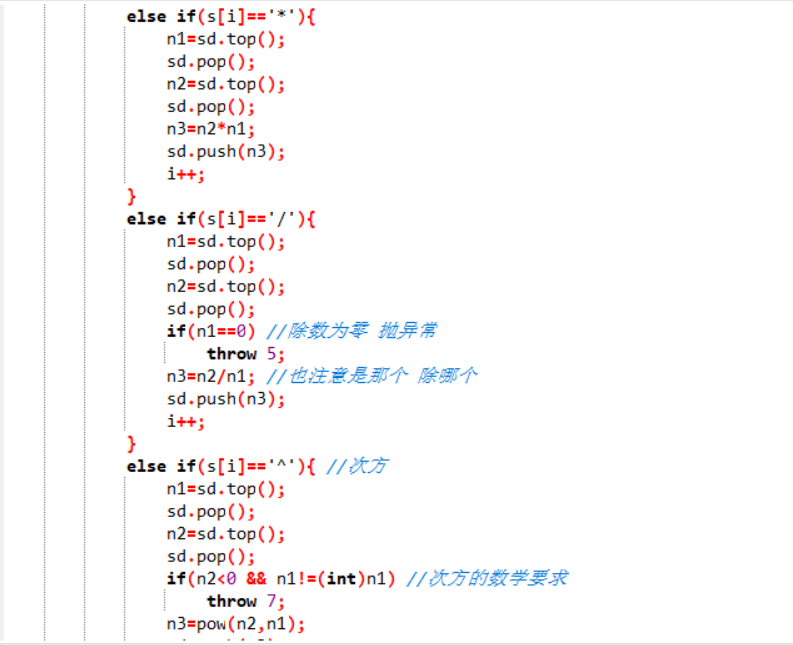


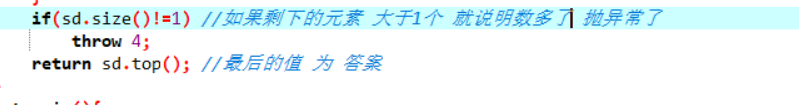


这整一个是关于将中缀表达式转化成逆波兰表达式的方法，转化成逆波兰表达式方便求值。其实里面主要涉及到的是两个循环了，一个是在数组的循环，也就是将s1（即中缀表达式）一直读到栈底，数字什么的入栈，另一个是当进入的是运算符的情况，将该字符与运算符栈顶的运算符的优先关系相比较。如果该字符优先关系高于此运算符栈顶的运算符，则将该运算符入栈。若不是的话，则将栈顶的运算符从栈中弹出，直到栈项运算符的优先级低于当前运算符，将该字符入栈。









转化为逆波兰表达式后求值就相对容易，比较烦的可能还是对于字符串转数字的方面了，也就是对于大于10的数字和有小数点的数字的输入。对于后面的计算，也就是在栈顶抛出两个元素进行运算后再入栈了，最后判断是否是只剩一个元素，来判断异常情况。

1. 程序运行方式简要说明

这个project感觉没有什么好说明的吧，就是输入一个新的算术表达式，然后输出结果了

