

程序设计 LAB 15

TA

张栩晨(09ss/13cs硕)

曾君(09ss/13ss硕)

刘骐硕(10ss)

感谢本学期代理TA

叶家杰(10ss)

张宓(09ss/13ss硕)

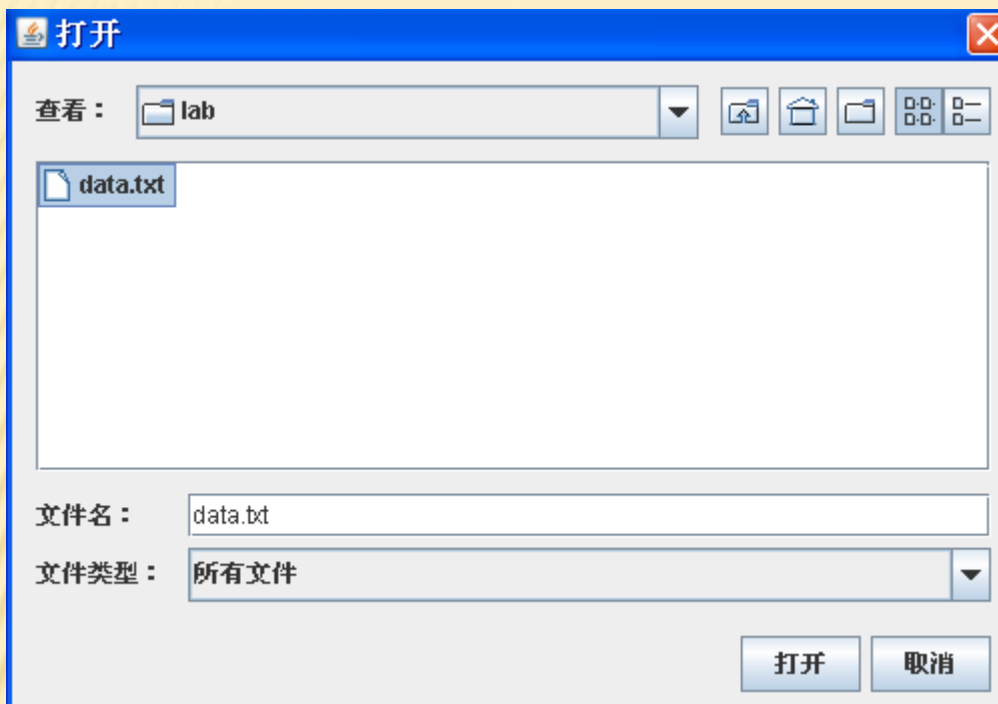
本次lab要求从文件中读出记录，根据记录构造表达式进行数学运算。同时，还要对出现异常的情况进行日志记录。

1) 实验中会给出一个 data.txt 的数据文件，文件中每一行为一条记录。每条记录就是一个形如(a \$ b)的计算表达式，操作数与运算符间用空格隔开，其中a、b为数字，\$为+ - * /运算符，合法的表达式长度为3。

注意：有一些表达式是不合法的，比如参数个数不够或过多、参数无法从字符串转换成数字、没有对应的计算操作等，要求每次读到一条不合法的记录都要将相应的异常抛出并用日志记录异常信息。

要求窗口程序要以文件浏览的方式(可以用JFileChooser)将 data.txt读入。

```
1 + 1
2 * 4
5 - 5
1
aaaa + 87
65 kkk 65
```



执行程序后，将合法算式和结果
写回源文件

```
1 + 1=2.0  
2 * 4=8.0  
5 - 5=0.0  
87 * 45=3915.0  
76 / 8=9.5  
45 - 14=31.0  
88 - 66=22.0  
34 * 67=2278.0  
35 + 76=111.0
```


2) 处理各条记录时，根据情况抛出不同类型的异常。外部程序再根据不同的异常进行日志记录等操作。在调用execute的函数对每个Exception调用对应的log()方法。

```
public void execute(String record) throws OperatorException, DivedByZeroException,
    NoNumberException, ParamTooManyException, ParamNotEnoughException {
    try {
        //do something
        //throw new ParamTooManyException(record + "参数过多", log);
        //throw new OperatorException(record + "操作符错误", log);
    } catch (ParamTooManyException ptm) {
        throw ptm;
    } catch (ArrayIndexOutOfBoundsException notenough) {
        throw new ParamNotEnoughException(record + "参数过少", log);
    } catch (NumberFormatException forexception) {
        throw new NoNumberException(record + "无法转成数字", log);
    } catch (ArithmeticException aexception) {
        throw new DivedByZeroException(record + "除数为零", log);
    } catch (OperatorException oexception) {
        throw oexception;
    }
}
```

你需要自己定义以下几个异常类：

```
public class NoNumberException extends NumberFormatException(字符串无法转换成数字)
```

```
public class ParamNotEnoughException extends  
ArrayIndexOutOfBoundsException ( 参数过少 )
```

```
public class DivedByZeroException extends ArithmeticException ( 除零 )
```

```
public class ParamTooManyException extends Exception (参数过多)
```

```
public class OperatorException extends Exception ( 操作符有误 )
```

以ParamTooManyException (参数过多)为例，该类继承自Exception类，并提供了一个log()的方法用于日志记录

```
public class ParamTooManyException extends Exception {  
    private MyLogger mylog;  
    public ParamTooMore(String message, MyLogger mylog) {  
        super(message);  
        this.mylog = mylog;  
    }  
    public void log() {  
        //往日志中添加内容  
        log.warning(getMessage());  
    }  
}
```

3) 日志记录

自己实现一个MyLogger类并且提供warring()方法，将异常记录信息写入另一个文件，文件名为output.txt

这是本学期最后一堂lab课， Byebye !

请提交整个项目， **deadline: 2013.1.5 晚23点**

遇到bug时，请牢记使用**加输出语句**，**单步调试**来debug。
程序逻辑的错误bug是一定可以de出来的，其他bug无法解决
请先去搜索引擎查找答案，之后再请教他人。
请相信没有bug是de不出来的。