一、实验目的

□ □ □ □ □ □ □ □ □ □

实验二 □自动化测试

班级	学号	姓名	指导老师
软件工程□ □ □ □班	□ □ □ □ □ □ □ □ □ □	董佩杰	毛锐

一、实验目的

□ □ 掌握□ □ □ □ □的安装及其与□ □ □ □ □ □ □的集成。

□ □ 利用□ □ □ □ □进行单元测试。

□ □ 掌握□ □ □ □ □中常用□ □ □ □ □ □ □ □ □ □：□ □ □ □ □ □ □、□ □ □ □ □ □、□ □ □ □ □、□ □ □ □ □ □ □

□ □ 掌握□ □ □ □ □中套件测试和参数化测试的方法。

□ □ 掌握□ □ □ □ □ □ □的安装和使用。

□ □ 基于□ □ □ □ □ □ □的覆盖率测试对□ □ □ □ □单元测试覆盖分析，提升测试质量。

□ □ 利用□ □ □进行自动化测试的配置和执行

二、实验步骤

□ □ □ □ □ □ □的安装及其与□ □ □ □ □ □ □的集成。

□两种方法，□ □ □ □ □ □ □本地安装，□ □ □ □ □ □ □ □ □集成安装□

□ □ □从□ □ □ □ □ □ □ □ □ □ □ □ □下载□ □ □ □ □压缩包，把□ □ □ □ □压缩包解压到一个物理路径

□ □ □记录□ □ □ □ □ □ □ □ □ □ □ □文件所在目录。

□ □ □设置环境变量□ □ □ □ □ □ □ □ □。

□ □ □在□ □ □ □ □ □ □菜单□ □ □ □ □ □ □ □ □的子项□ □ □ □ □ □ □ □ □ □ □中选择□ □ □ □ □
□ □ □ □ □ □ □ □，单击□ □ □ □ □ □ □ □ □ □标签，添加□ □ □ □，即选择□ □ □ □ □ □ □ □ □或□ □ □ □ □ □
□ □ □ □ □ □ □ □，单击打开，就完成了□ □ □ □ □的安装。

二、实验步骤

[illegible]单元测试

[illegible]实习题一

利用[illegible]生成测试用例的框架，在框架中设计测试代码，完成对下面类[illegible]

[illegible]求解最大公约数和最小公倍数以及查找功能

[illegible]求最大公约数

[illegible]最小公倍数

[illegible]查找

[illegible]

二、实验步骤

主函数

最大公约数为：

最小公倍数为：

查找结果为：

测试类：

对

进行测试

二、实验步骤

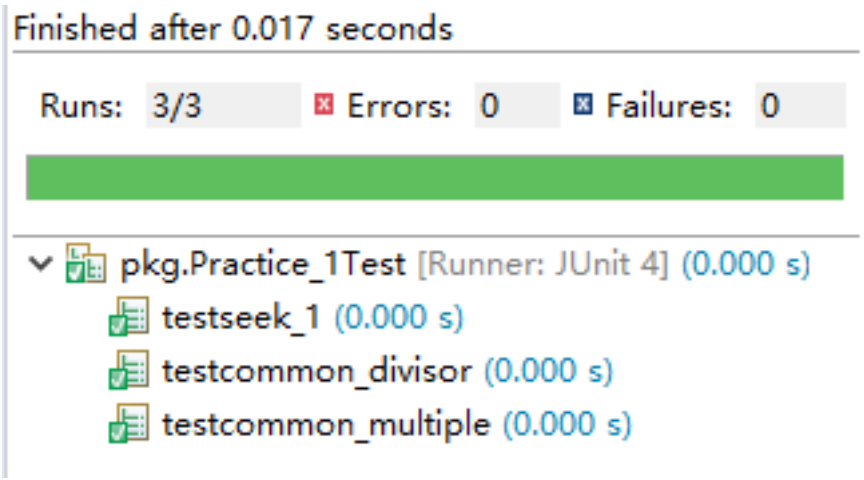
测试结果：

实习题二

设计判断一个数是不是素数的程序，用基本断言类型实现测试，并用初始化测试环境。参数化测试数据

判断素数类：

二、实验步骤



判断一个证书是否为素数

二、实验步骤

参数化测试，测试素数

二、实验步骤

测试结果：

使用 对以上两个测试类进行测试：

测试结果为：

二、实验步骤


Finished after 0.026 seconds
Runs: 18/18
Errors: 0
Failures: 0
pkg2.PrimeTest [Runner: JUnit 4] (0.001 s)
[0] (0.000 s)
[1] (0.000 s)
[2] (0.000 s)
[3] (0.000 s)
[4] (0.000 s)
[5] (0.000 s)
[6] (0.000 s)
[7] (0.000 s)
[8] (0.000 s)
[9] (0.000 s)
[10] (0.000 s)
[11] (0.000 s)
[12] (0.000 s)
[13] (0.000 s)
[14] (0.000 s)
[15] (0.000 s)
[16] (0.000 s)
[17] (0.000 s)

二、实验步骤


pkg2.AllTest [Runner: JUnit 4] (0.001 s)
pkg2.PrimeTest (0.001 s)
[0] (0.000 s)
[1] (0.000 s)
[2] (0.001 s)
[3] (0.000 s)
[4] (0.000 s)
[5] (0.000 s)
[6] (0.000 s)
[7] (0.000 s)
[8] (0.000 s)
[9] (0.000 s)
[10] (0.000 s)
[11] (0.000 s)
[12] (0.000 s)
[13] (0.000 s)
[14] (0.000 s)
[15] (0.000 s)
[16] (0.000 s)
[17] (0.000 s)
pkg.Practice_1Test (0.000 s)
testseek_1 (0.000 s)
testcommon_divisor (0.000 s)
testcommon_multiple (0.000 s)

二、实验步骤

实习题三

下面是使用来跟踪一年中的那些天是节假日的程序。

判断一个数是否存在于这个列表

集合中假日是随机设定的，可根据今年的情况自行调整

二、实验步骤

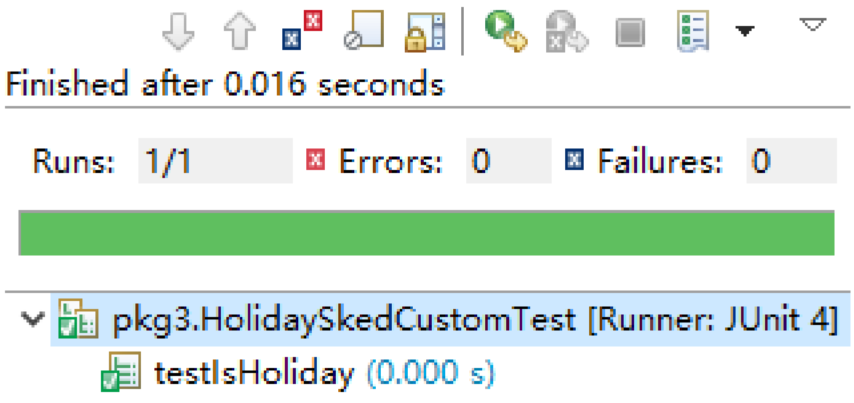
请用方法对程序中的方法进行测试；

用方法对程序中的方法进行测试

二、实验步骤

□

测试结果：



□□□□□□ □□ □□□□□□□□□□□□□

□□□ 用参数化的方法重新设计本题和实验题□的测试用例。

本题测试用例：

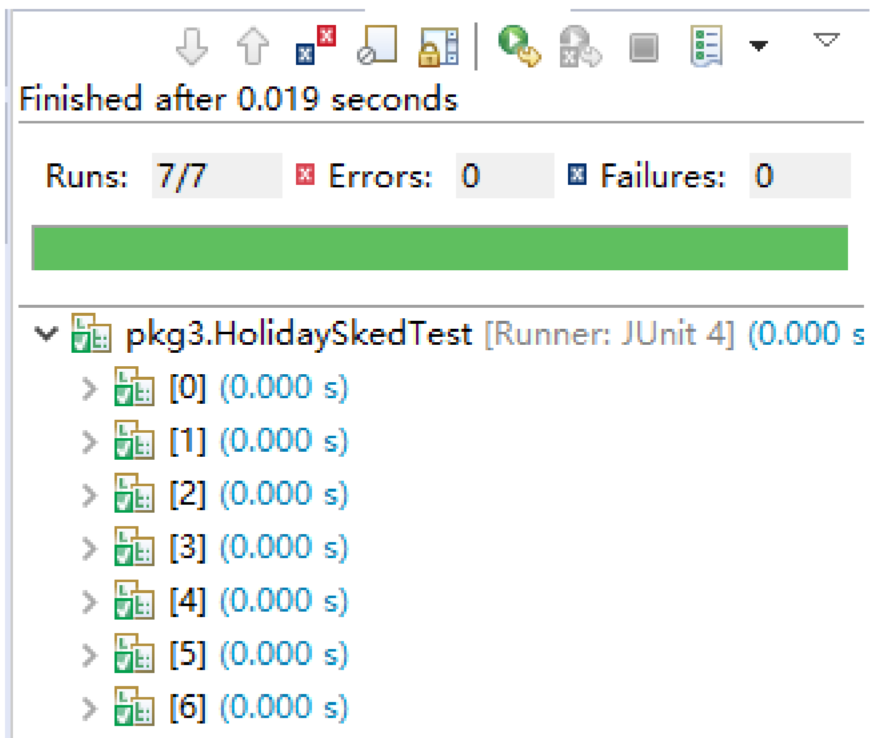
```
□□□□□□□ □□□□□

□□□□□□ □□□□□□ □□□□□□□□□□□□□□□□□□□

□□□□□□ □□□□□□□□□□□□□□□□□
□□□□□□ □□□□□□□□□□□□□□□□□□□□□

□□□□□□ □□□□□□□□□□□□□□□
□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

□□□
 □ □□□□□□□ □□□□
 □ □□□□□□□□□用参数化方法进行测试
 □□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□ □□□□□ □□□□□□□□□□□□□□□ □
```

二、实验步骤

实验□测试用例：□由于参数的异构，所以需要分为两个部分进行测试□

第一部分：最大公约数和最小公倍数

二、实验步骤


Finished after 0.019 seconds
Runs: 7/7
Errors: 0
Failures: 0
pkg3.HolidaySkedTest [Runner: JUnit 4] (0.000 s
[0] (0.000 s)
[1] (0.000 s)
[2] (0.000 s)
[3] (0.000 s)
[4] (0.000 s)
[5] (0.000 s)
[6] (0.000 s)

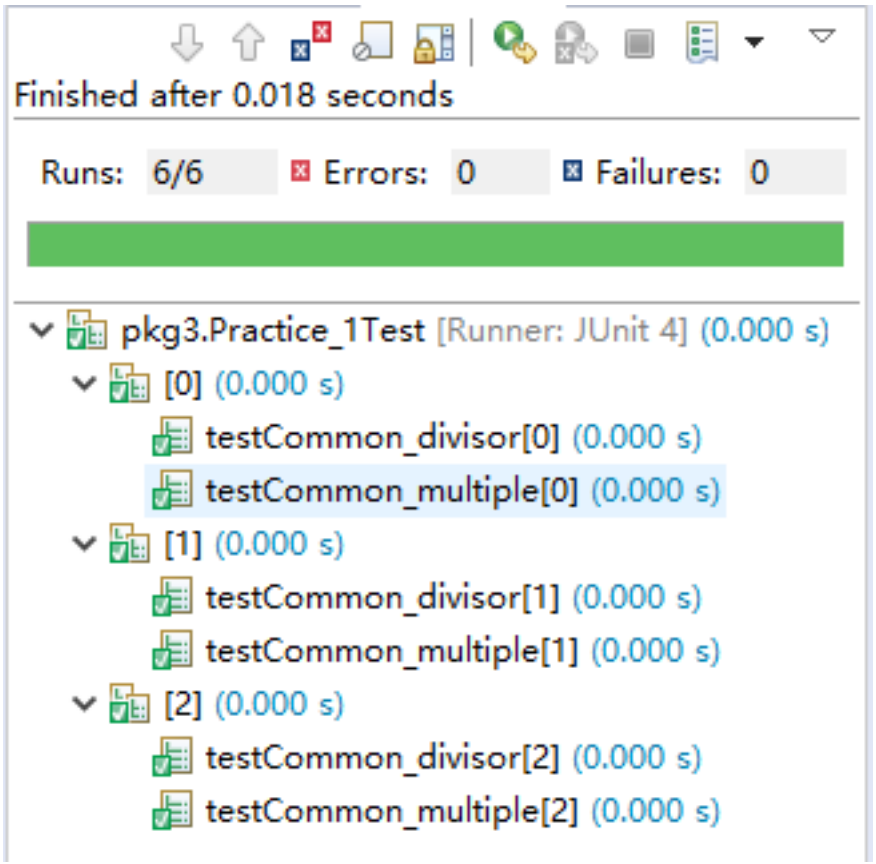
二、实验步骤

主要用来测试 和 两个方法

公倍数

二、实验步骤

测试结果：

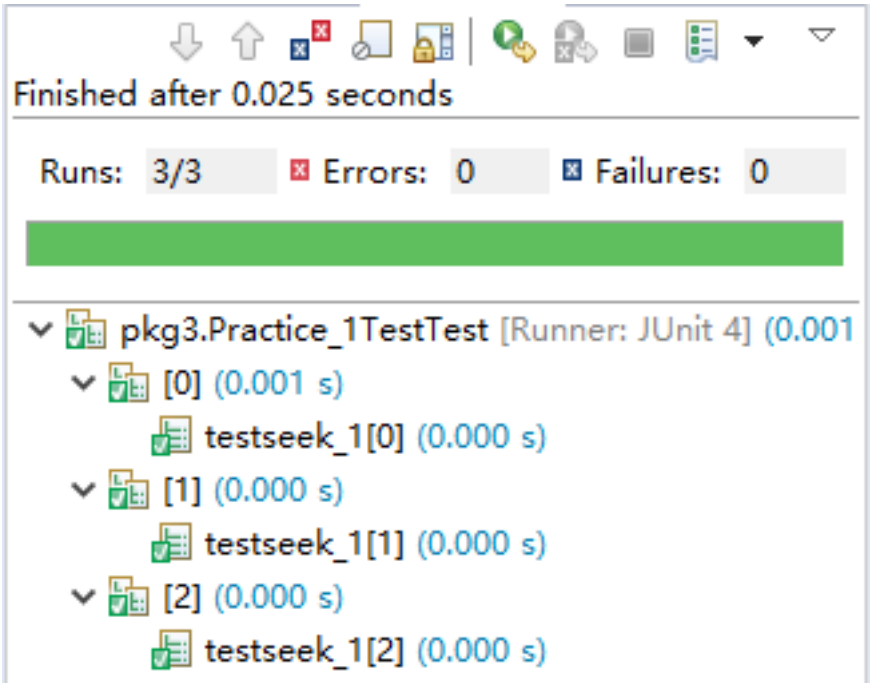

Finished after 0.018 seconds
Runs: 6/6
Errors: 0
Failures: 0
pkg3.Practice_1Test [Runner: JUnit 4] (0.000 s)
[0] (0.000 s)
testCommon_divisor[0] (0.000 s)
testCommon_multiple[0] (0.000 s)
[1] (0.000 s)
testCommon_divisor[1] (0.000 s)
testCommon_multiple[1] (0.000 s)
[2] (0.000 s)
testCommon_divisor[2] (0.000 s)
testCommon_multiple[2] (0.000 s)

二、实验步骤

第二部分：查找功能

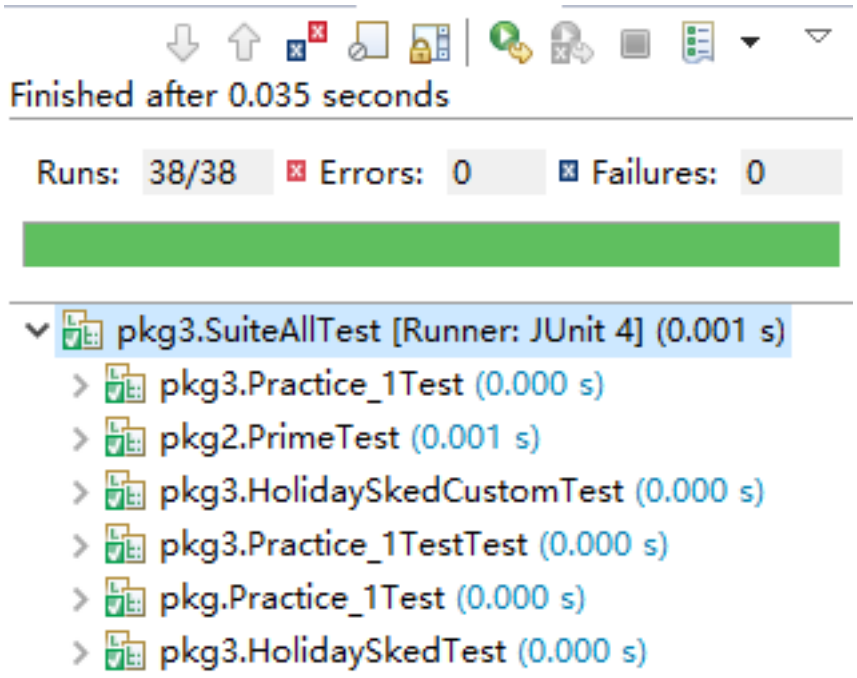
二、实验步骤

测试结果：


Finished after 0.025 seconds
Runs: 3/3
Errors: 0
Failures: 0
pkg3.Practice_1TestTest [Runner: JUnit 4] (0.001
[0] (0.001 s)
testseek_1[0] (0.000 s)
[1] (0.000 s)
testseek_1[1] (0.000 s)
[2] (0.000 s)
testseek_1[2] (0.000 s)

二、实验步骤

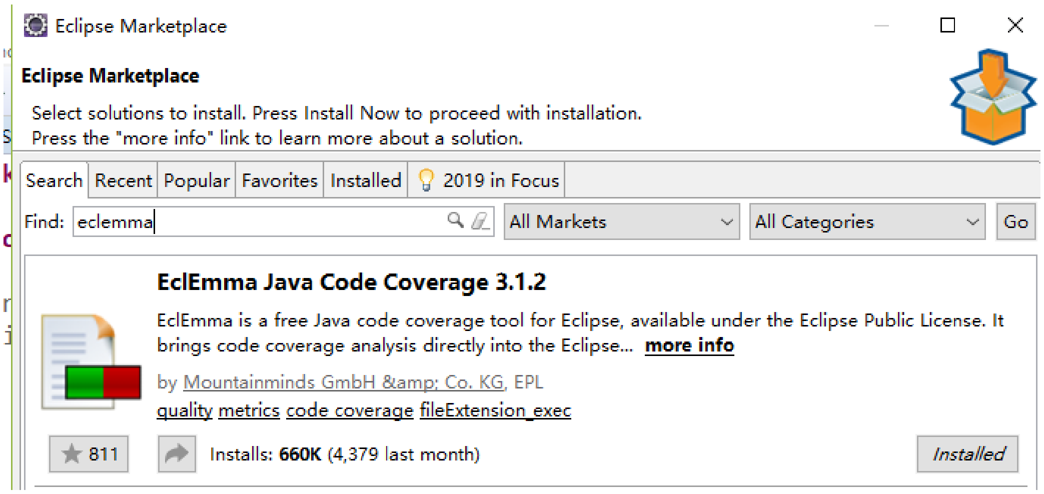
测试结果：


Finished after 0.035 seconds
Runs: 38/38
Errors: 0
Failures: 0
pkg3.SuiteAllTest [Runner: JUnit 4] (0.001 s)
pkg3.Practice_1Test (0.000 s)
pkg2.PrimeTest (0.001 s)
pkg3.HolidaySkedCustomTest (0.000 s)
pkg3.Practice_1TestTest (0.000 s)
pkg.Practice_1Test (0.000 s)
pkg3.HolidaySkedTest (0.000 s)

二、实验步骤

□ □ 实习题四

对课本 □ □ □ 页的示例程序利用语句覆盖、判定 □ 条件覆盖、条件组合及路径覆盖的角度分别设计测试用例进行自动化



□ □

已安装好 □ □ □ □ □ □ □

□ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □

□ □ 实习题五

针对静态测试试验中选择排序，三角形问题，隔一日问题的代码，或者自己开发实现的综合性 □ □ □ □ 项目，在所学测

针对每一个类使用 □ □ □ □ □ 进行自动化测试，然后使用 □ □ □ □ □ 方法调用所有单个测试类，以下是执行的结果：

代码见附件

□ □ 实习题六

针对实验 □ 的 □ □ □ □ 项目，利用 □ □ □ 结合 □ □ □ □ □ 进行自动化测试构建运行。

通过使用 □ □ □ □ □ □ 将 □ □ □ □ □ □ □ □ □ 文件生成，然后运行即可。

在这个过程中遇到了一个比较大的问题，一开始将这些文件组织到一个新的工程中的时候，选择将其复制到工程中，然

二、实验步骤

DateProcess	100.0 %	65	0	65
judge(int[])	100.0 %	26	0	26
myGetNextDate(int[])	100.0 %	36	0	36
DateProcessTest	100.0 %	153	0	153
setUp()	100.0 %	6	0	6
test()	100.0 %	8	0	8
testJudge()	100.0 %	37	0	37
SelectionSort	100.0 %	54	0	54
selectionSort(int[])	100.0 %	51	0	51
SelectionSortTest	100.0 %	72	0	72
setUp()	100.0 %	6	0	6
test()	100.0 %	15	0	15
Triangle	100.0 %	142	0	142
judge(int[])	100.0 %	37	0	37
judgeTriangle(int[])	100.0 %	51	0	51
selectionSort(int[])	100.0 %	51	0	51
TriangleTest	100.0 %	135	0	135
setUp()	100.0 %	6	0	6
test()	100.0 %	33	0	33
testJudge()	100.0 %	7	0	7
testSort()	100.0 %	7	0	7

□ □

```
Buildfile: E:\JavaSpace\exp3_pkg6\build.xml
DateProcessTest (1):
    [junit] Running pkg6_test.DateProcessTest
    [junit] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.109 sec
SelectionSortTest (1):
    [junit] Running pkg6_test.SelectionSortTest
    [junit] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.061 sec
TriangleTest (1):
    [junit] Running pkg6_test.TriangleTest
    [junit] Tests run: 3, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.061 sec
TestSuites (1):
    [junit] Running pkg6_test.TestSuites
    [junit] Tests run: 6, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.069 sec
BUILD SUCCESSFUL
Total time: 3 seconds
```

□ □

三、 总结

本次实习经历的时间比较长，前半部分做的比较快，后半部分由于一些环境配置还有其他奇奇怪怪的原因导致了代码无