****

**软件测试技术第四次实验报告**



**学 院 软件学院**

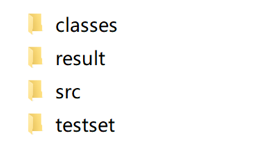
**专 业 软件工程**

**年 级 2016级**

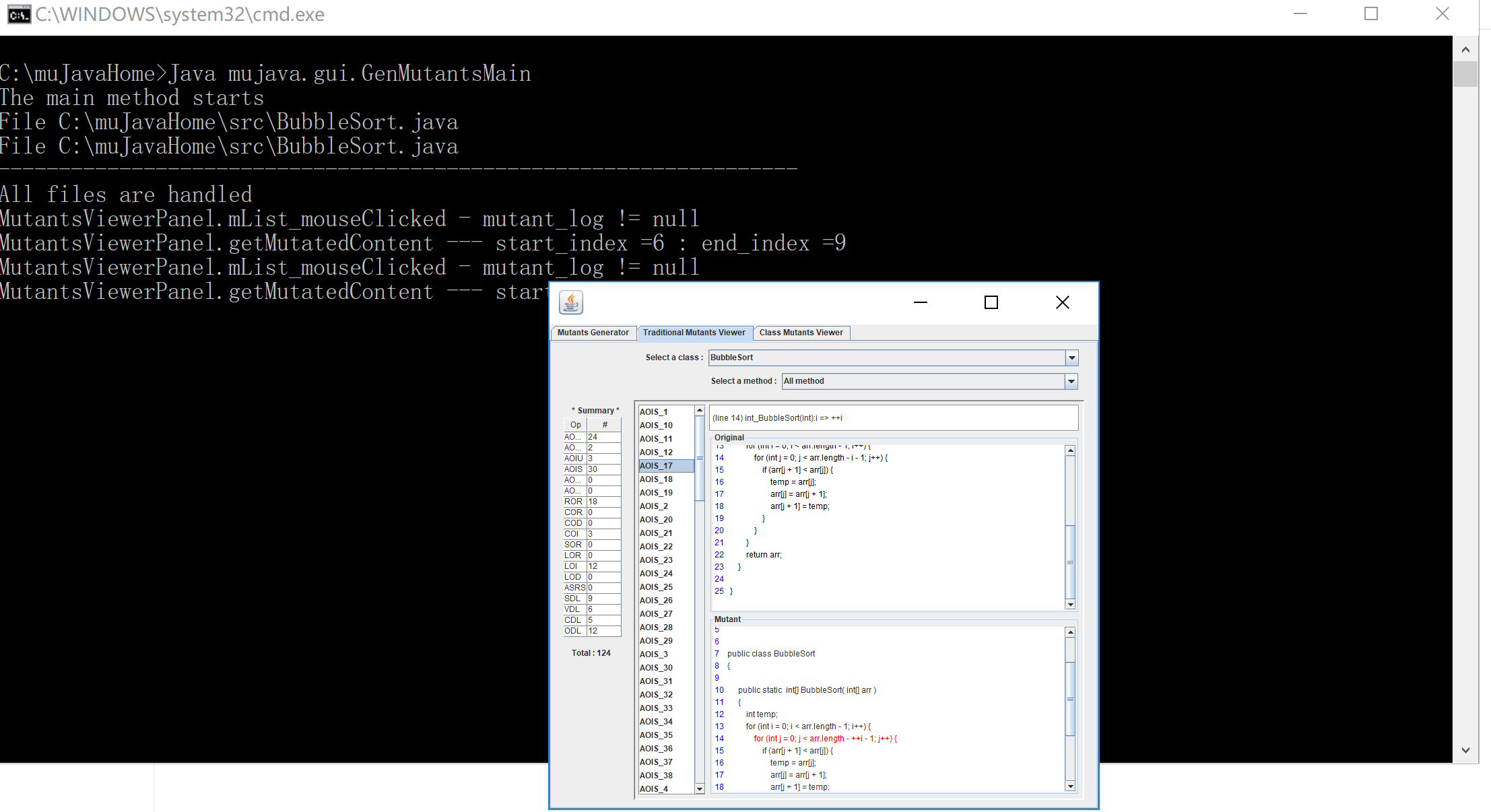
**姓 名 霍欣芷**

**2019年 4 月 18 日**

# 软件测试技术第四次实验报告

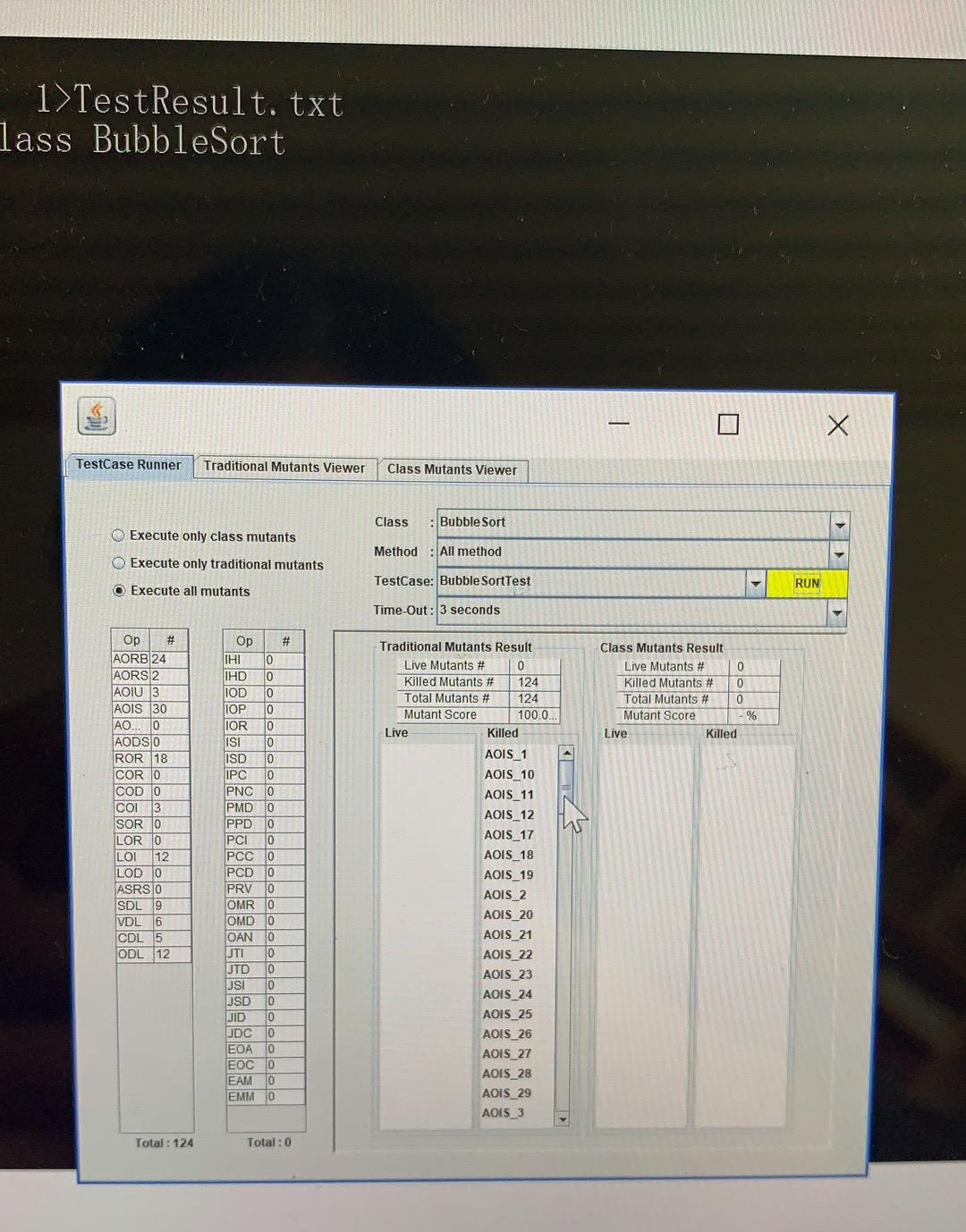
1. 需求分析（描述具体需求）
2. Install MuJava. The instruction of how to install and use Mujava can be seen in <https://cs.gmu.edu/~offutt/mujava/> .
3. Two small programs are given for your task. BubbleSort.java is an implementation of bubble sort algorithm and BackPack.java is a solution of 01 backpack problem. Try to generate Mutants of 2 given programs with MuJava.
4. Write testing sets for 2 programs with Junit, and run mutants on the test sets with MuJava.
5. 概要设计（简单描述设计思路，配合UML图）
6. 将mujava.jar、openjava.jar、mujava.config、junit.jar 下载下来
7. 将下载的mujava相关文件放到一个文件夹并添加mujava环境变量
8. 创建一个文件夹，C:\mujavaHome；将mujava.config 移至该文件夹，并将MuJava\_HOME设置为该文件夹，最后在该文件夹下创建如下结构目录：   
   
9. 在C:\mujavaHome下创建两个Txt，并另存为cmd后缀，方便运行mujava程序。

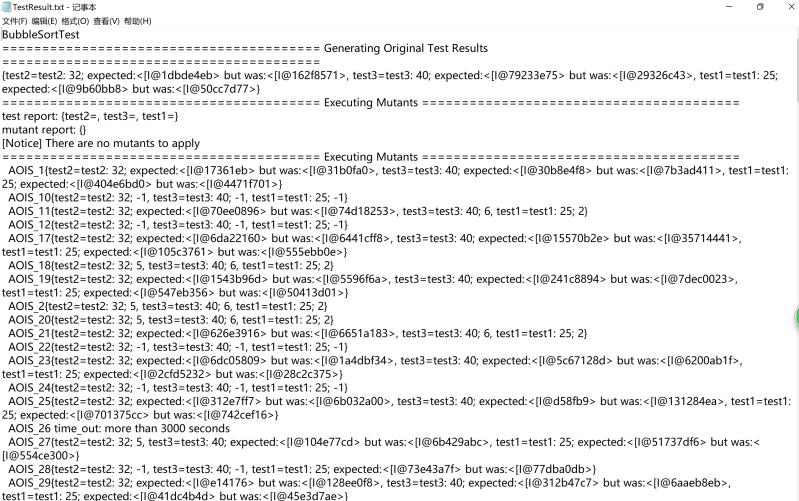
5、将BubbleSort.java复制到src文件夹，将编译生成的BubbleSort.class复制到classes文件夹，运行。



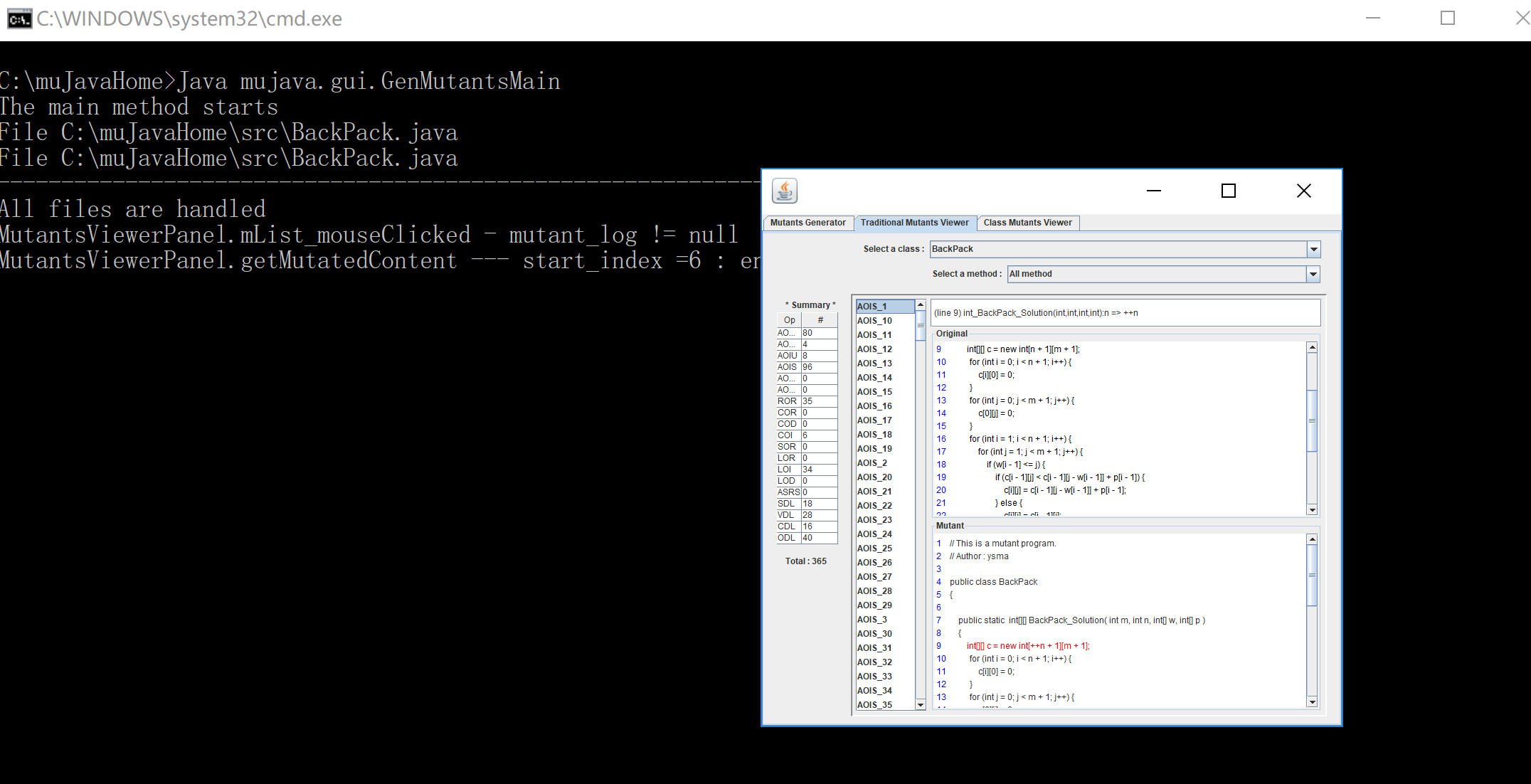
* 编写测试用例，将JUnit测试用例的.class文件复制在testset文件夹。

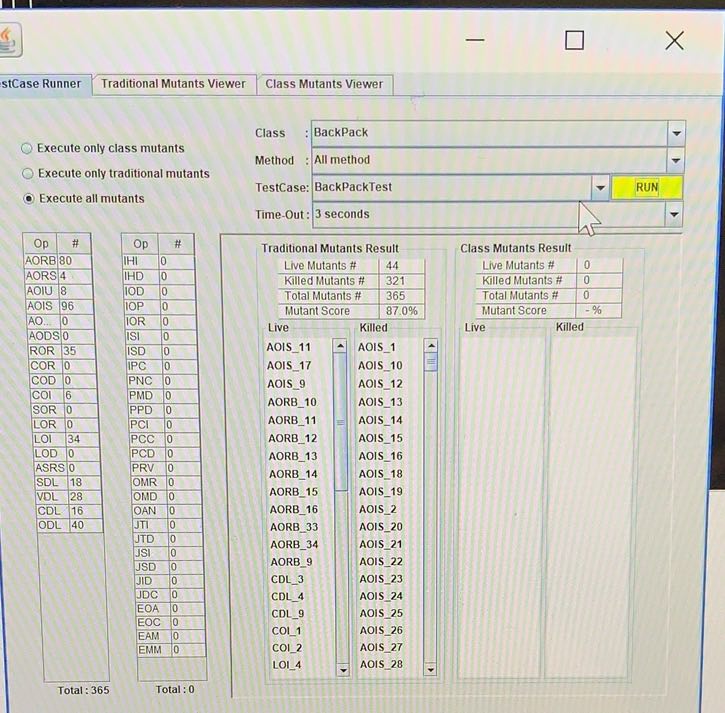
运行RunTest.cmd，完成对Calculator.java的变异测试。





对BackPack.java进行变异测试





1. 详细设计（详细描述具体如何实现，附代码及说明）

import static org.junit.Assert.\*;

import org.junit.After;

import org.junit.Before;

import org.junit.Test;

public class BubbleSortTest {

private BubbleSort c ;

@Before

public void setUp() throws Exception {

bs = new BubbleSort();

}

@After

public void tearDown() throws Exception {

bs = null;

}

@Test

public void test1() {

int []a = {1,0};

int []b = {0,1};

assertEquals(a, bs.BubbleSort(b));

}

@Test

public void test2() {

int []a = {5，6，7，8，9};

int []b = {9，5，8，7，6};

assertEquals(a, bs.BubbleSort(b));

}

@Test

public void test3() {

int []a = {-10,0,2,7,9,10};

int []b = {0,-10,9,2,7,10};

assertEquals(a, bs.BubbleSort(b));

}

}

BackPackTest.java

import static org.junit.Assert.\*;

import org.junit.Test;

public class BackPackTest {

@Test

public void test() {

int m = 10;

int n = 3;

int w[] = {3, 4, 5};

int p[] = {4, 5, 6};

int c[][] = BackPack.BackPack\_Solution(m, n, w, p);

for (int i = 1; i <= n; i++) {

for (int j = 1; j <= m; j++) {

//System.out.print(c[i][j] + " ");

if (j == m && i==3) {

System.out.print(c[i][j]);

assertEquals(c[i][j],11);

}

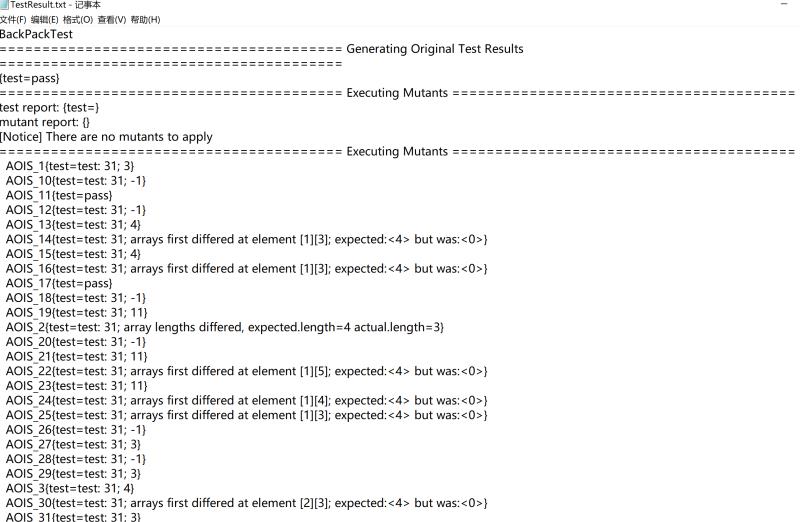
}

}

}

}

测试结果如下图所示：



1. 调试分析（在实验过程中遇到的问题以及如何解决）

在安装mujava时，发生了jdk版本不兼容的问题，将jdk版本调至1.8成功。

五、总结

了解了mujava的基本使用，又掌握了一项软件测试方法。同时再次熟悉了java路径配置的方法。