****

**软件测试技术第四次实验报告**



**学 院 智能与计算学部**

**专 业 软件工程**

**年 级 2016级**

**姓 名 许致远**

**2019年 4 月 25 日**

# 软件测试技术第四次实验报告

1. 需求分析：
2. 安装MuJava。 有关如何安装和使用Mujava的说明，请参见https://cs.gmu.edu/~offutt/mujava/。
3. 为您的任务提供两个小程序。 BubbleSort.java是一个冒泡排序算法的实现，BackPack.java是01背包问题的解决方案。 尝试使用MuJava生成2个给定程序的Mutants。

3.使用Junit编写2个程序的测试集，并使用MuJava在测试集上运行突变体。

1. 概要设计

Mujava的下载靠测试给的包，一共三个文件：junit.jar mujava.jar和openjava.jar。下载后创建一个文件，名为mujava。将三个文件放入，自建mujava.config、GenMutants.cmd、RunTest.cmd三个文件。创建四个文件夹名为：src result classes和testset.

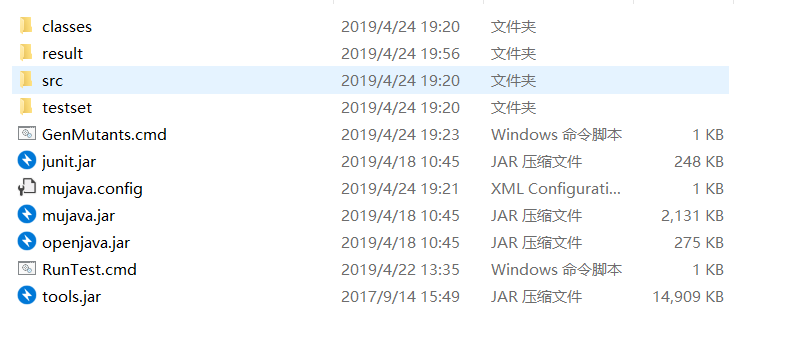
将测试的java文件放入src，将编译后的class文件放入classes，将写好的junit测试文件放入testset。

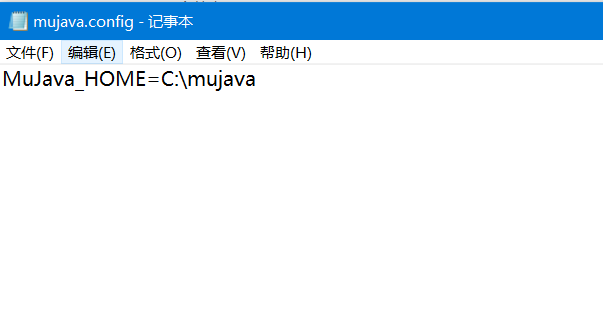
在控制面板中添加系统环境变量，新建变量名CLASSPATH，输入变量位置。

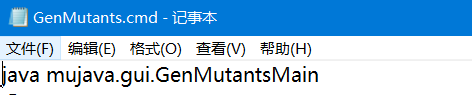
打开GenMutants.cmd即可开始生成变体，之后打开RunTest.cmd即可开始变异测试。

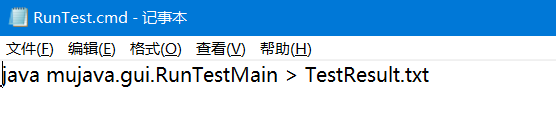
1. 详细设计

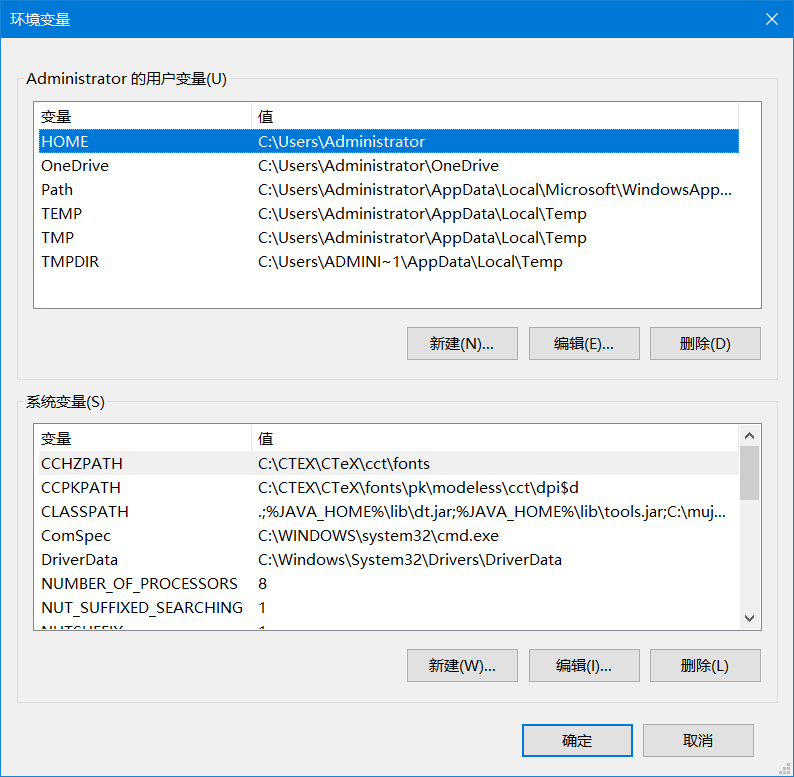
创建文件夹，放入相关的包，并创建相关路径，写相关测试用例：

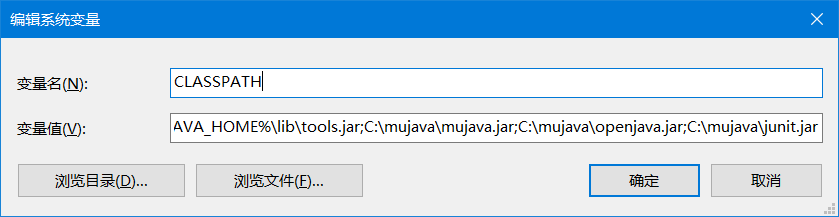




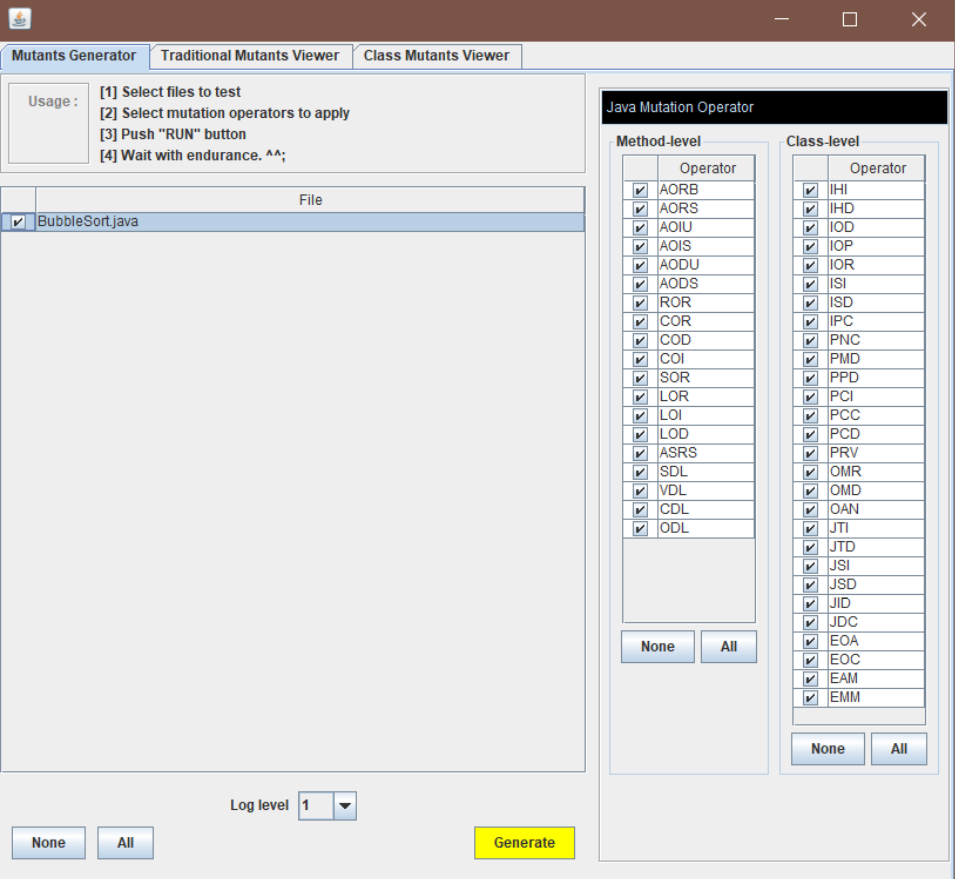


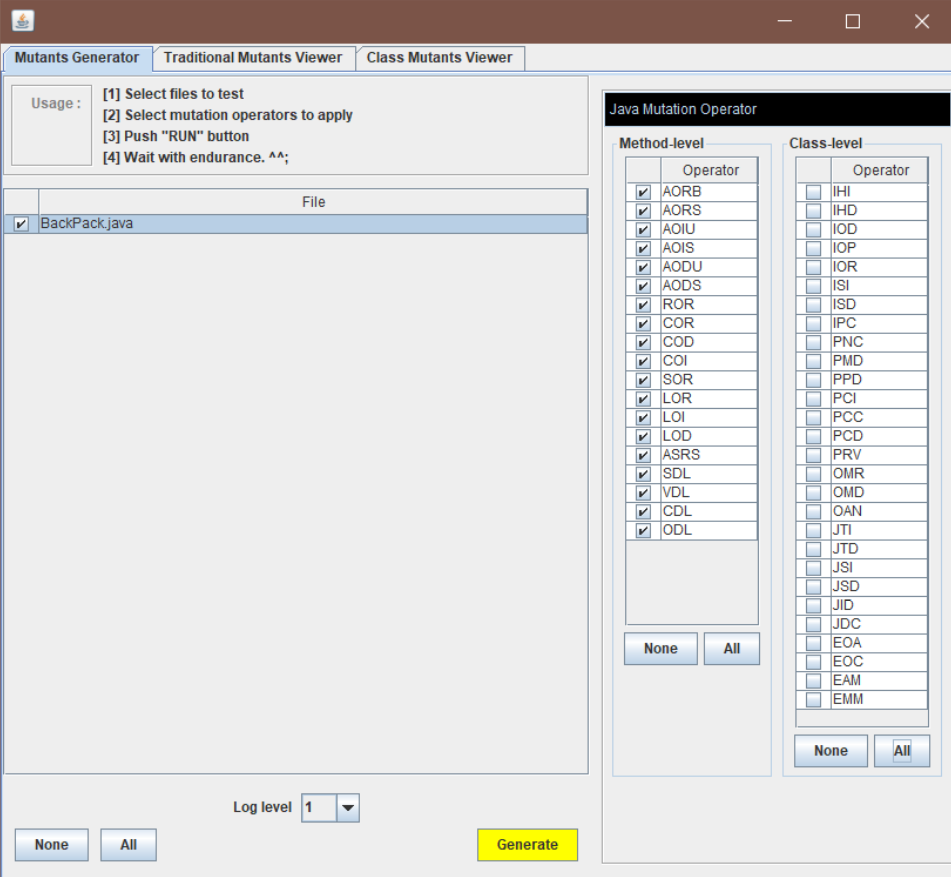




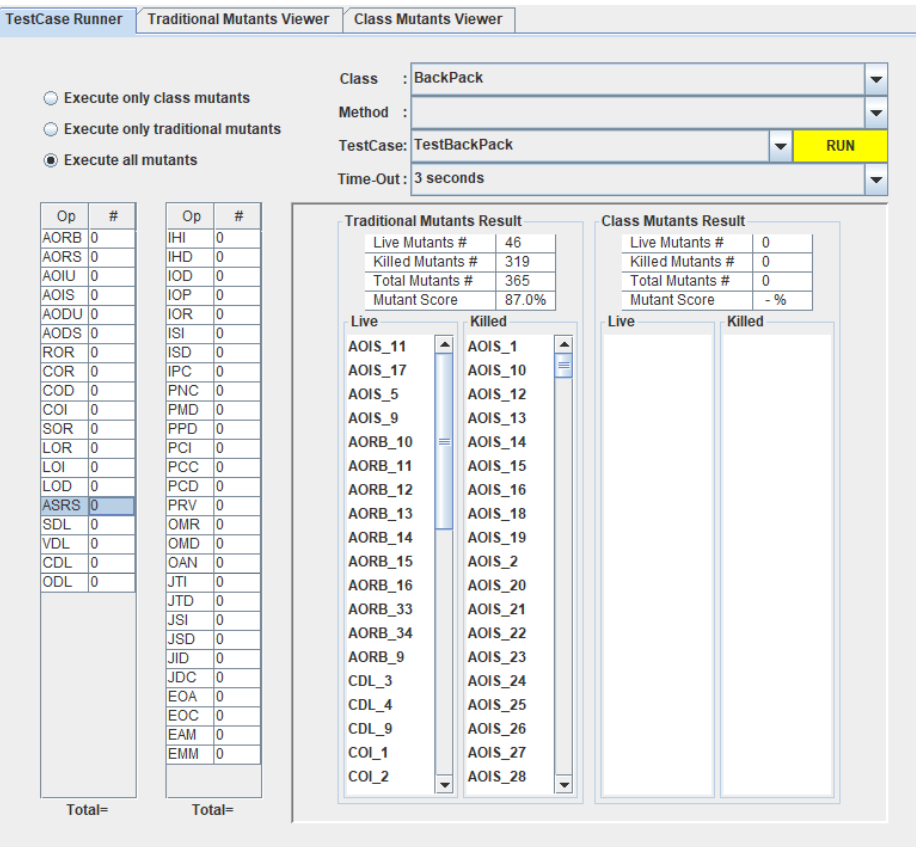


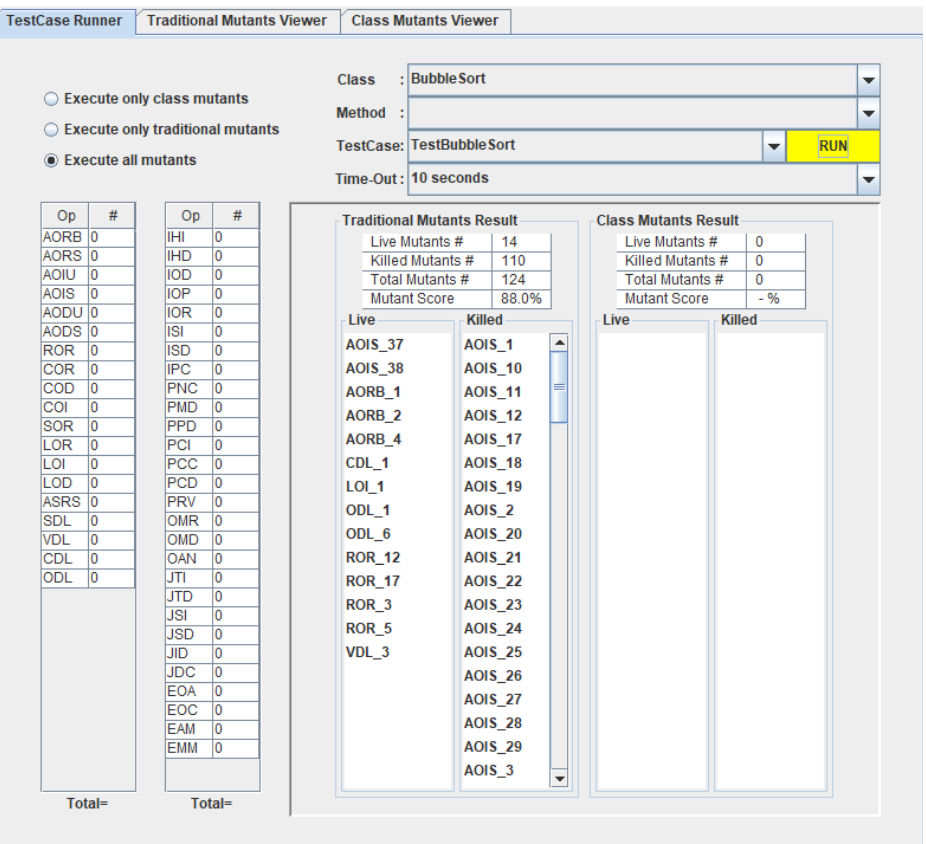
打开GenMutants.cmd，点击Generate开始生成变异体





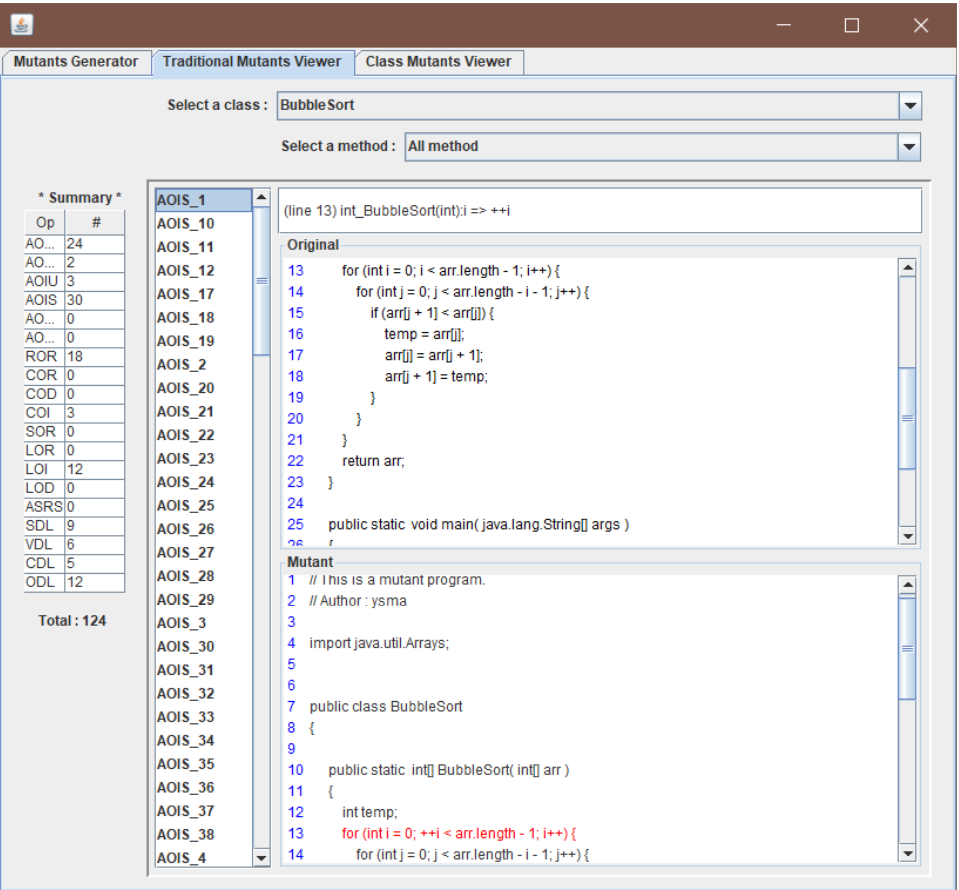
生成后再打开TestRun.cmd，点击RUN开始变异测试

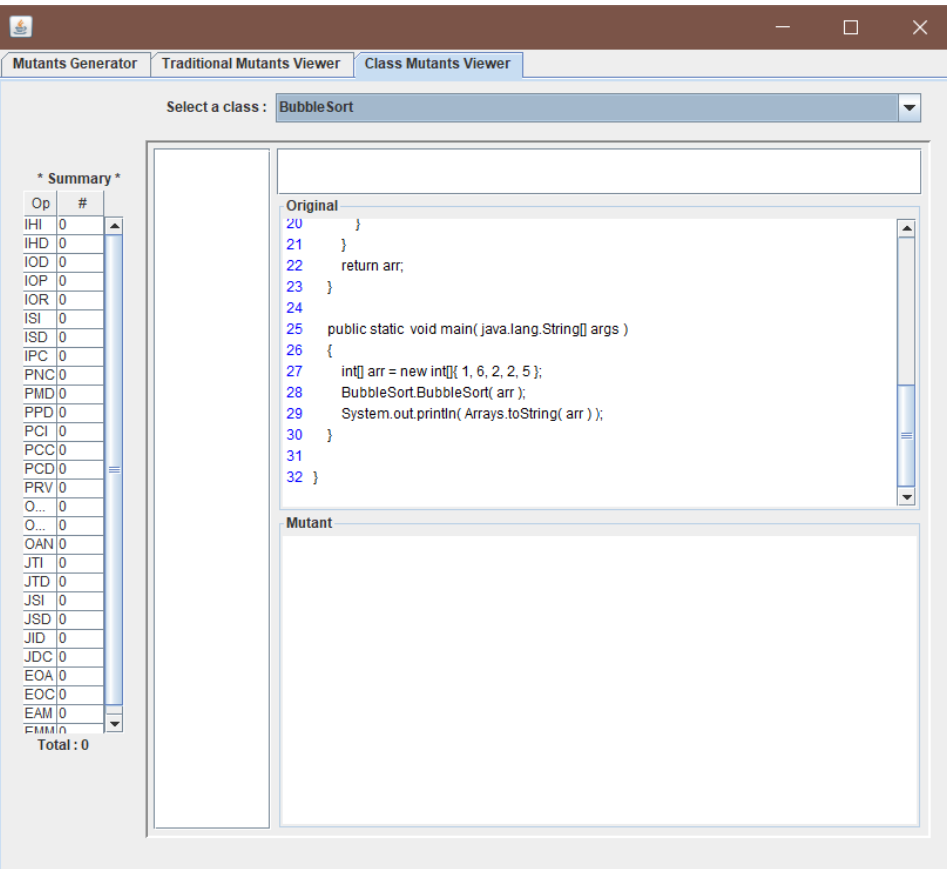




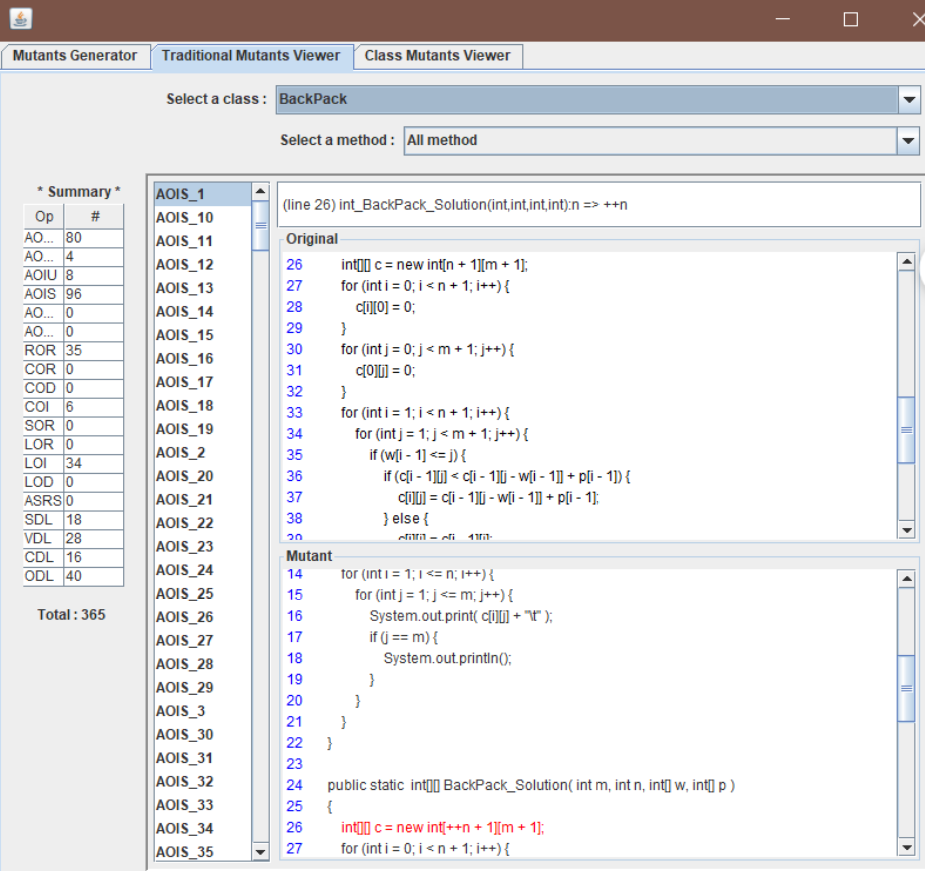
1. 测试结果

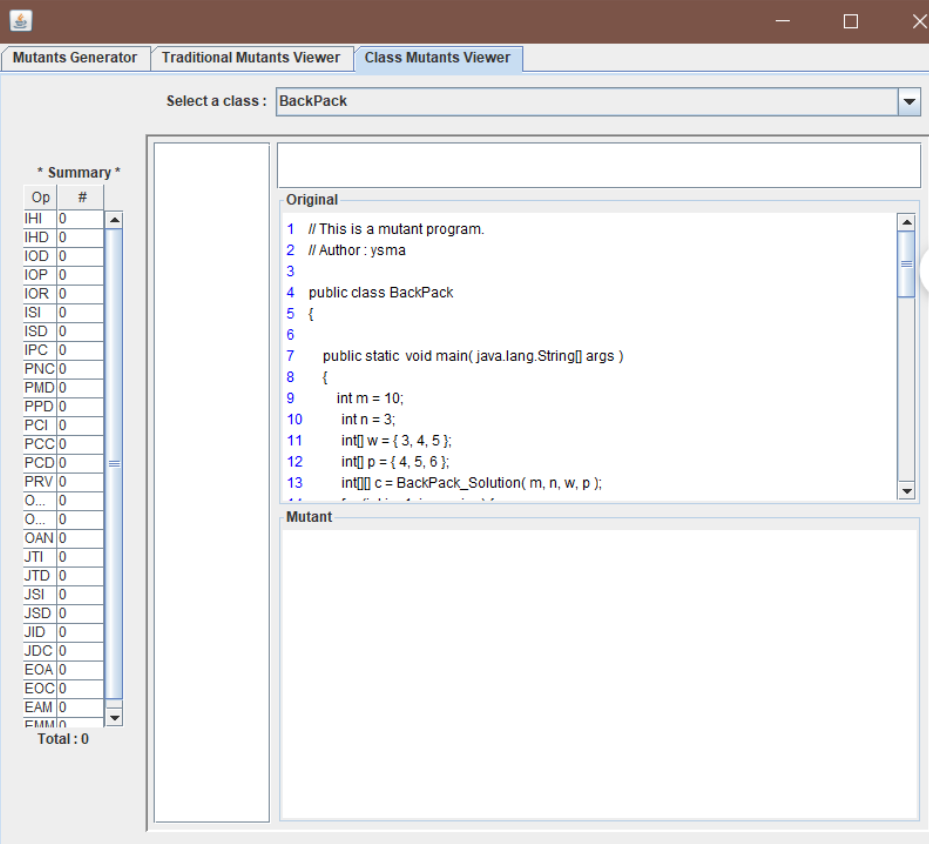
Bubble Sort：



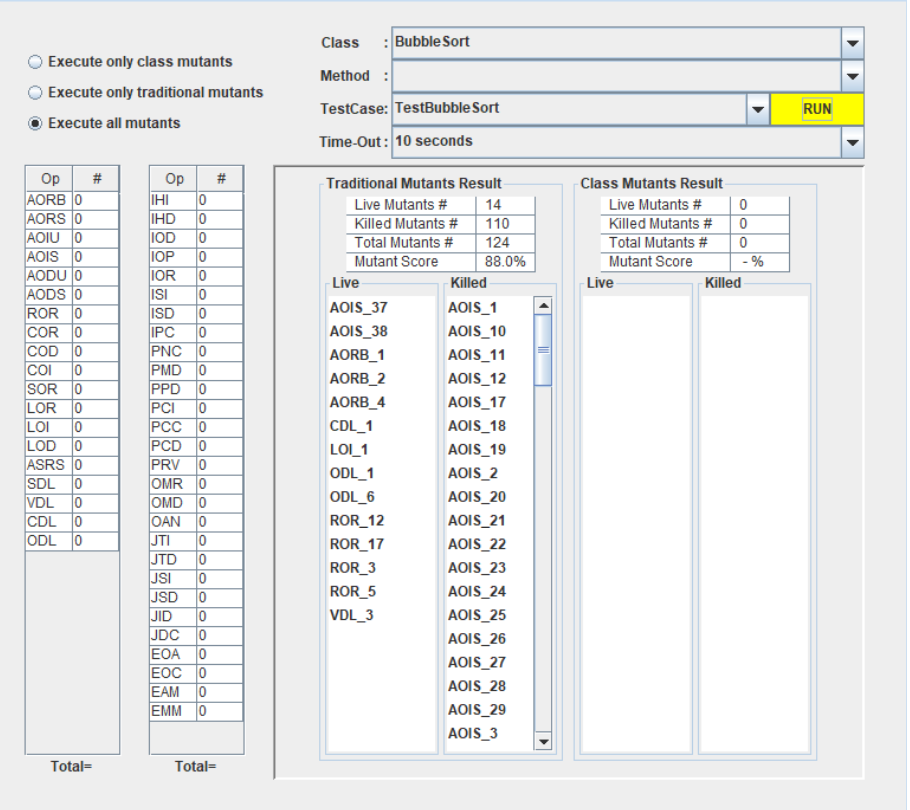


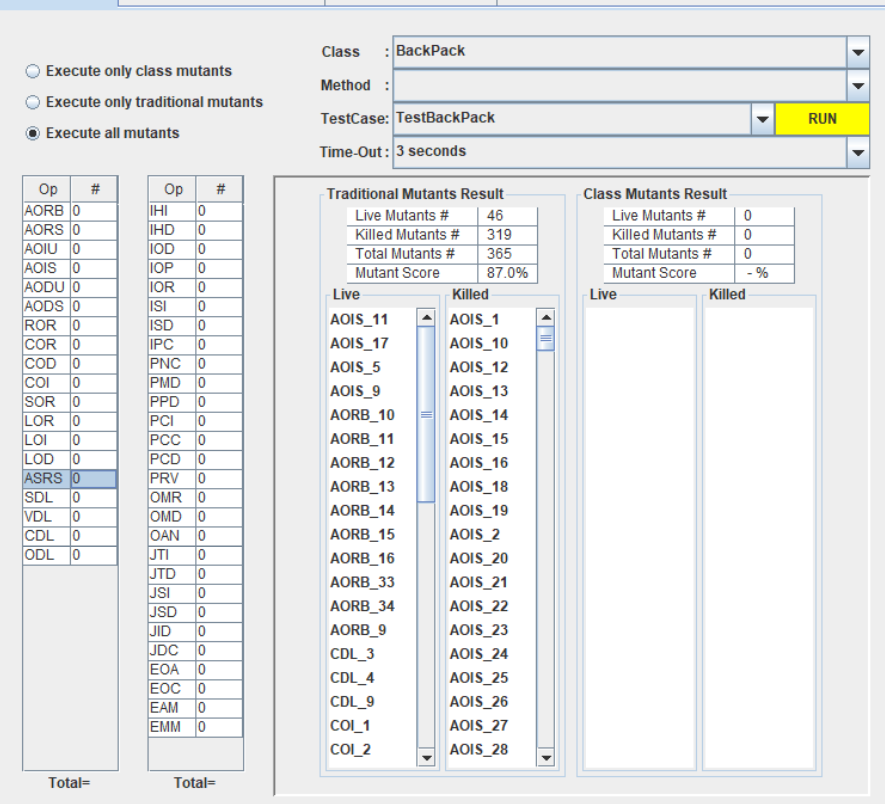
BackPack：





变异测试：





1. 总结

在一个工程中，潜在BUG的数量是巨大的，通过生成突变体来全面覆盖所有的错误是不可能的。所以我们需要使用变异测试去近似拟合相关的bug，使程序的错误尽可能的少，从而提高程序的鲁棒性和稳定性。这次实验使我学会了如何使用mujava进行变异测试，使我了解到如何去设计测试用例，并且使用mujava杀死变异体，受益匪浅。