2018全国大学生软件测试大赛

——开发者测试分项

全国大学生软件测试大赛组委会
2018年10月

MOOCTEST.org

概念介绍

开发者测试的概念

软件的持续快速迭代需求，大大压缩了软件开发的发布流程，使得一部分测试任务开始迁移，由软件开发人员担任这部分跟代码相关的软件测试工作，我们统称为开发者测试。开发者测试包括了传统的单元测试、集成测试、接口测试甚至部分系统测试相关的任务。

开发者需要对自己开发的程序代码承担质量责任。在软件质量管理机制下，一般要求开发者首先自行对自己编写的代码进行审查和测试，保证提交的代码必须达到一定的质量标准。开发者测试中的单元测试和集成测试，主要采用白盒测试方法，要求做测试的人员对软件代码非常熟悉。这样的测试任务，由软件开发人员开发者来做效率会更高。

概念介绍

分支覆盖

- **分支覆盖**要求程序中每个条件判定语句的真值结果和假值结果都至少出现一次
- 每个判断的真值结果和假值结果都至少出现一次相当于每个判断的真分支和假分支至少执行一次
- 分支覆盖不仅考虑了各个条件判定语句的覆盖需求，还考虑了这些语句分支的覆盖需求，因而较语句覆盖测试强度更高

概念介绍

- 程序 P1 包含 两条条件判定语句，这就要求相关的真假分支 ④、⑤、⑦、⑧ 至少被执行一次

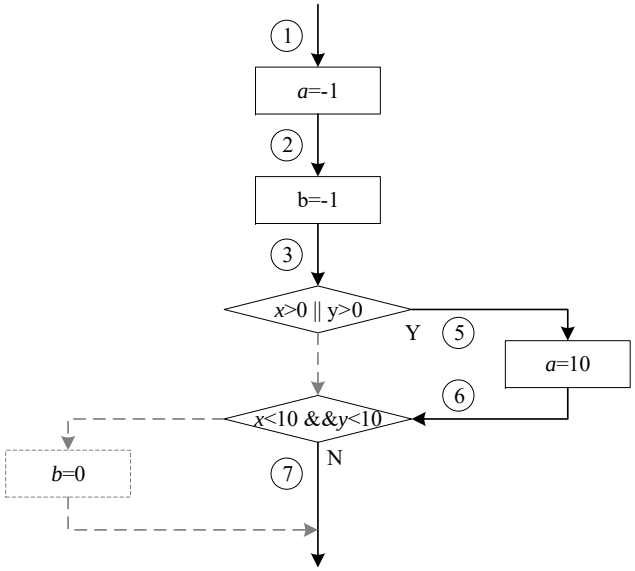


图 1(a) P1在t4=(20, 20)下的程序流程图

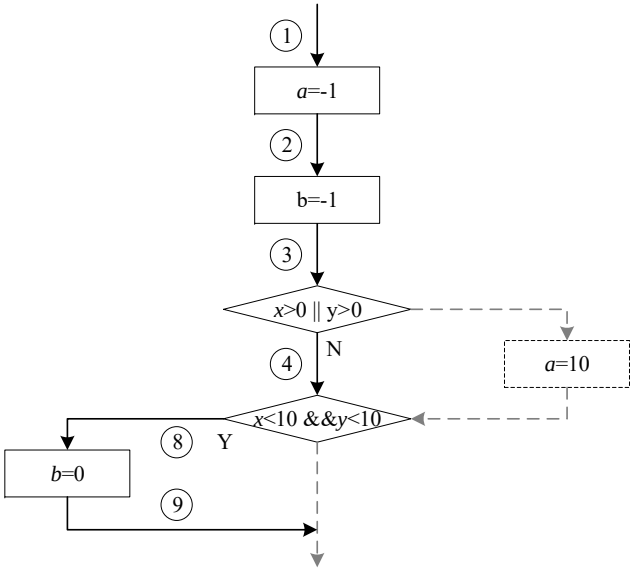


图 1(b) P1 在t5 =(-2, -2)下的程序流程图

概念介绍

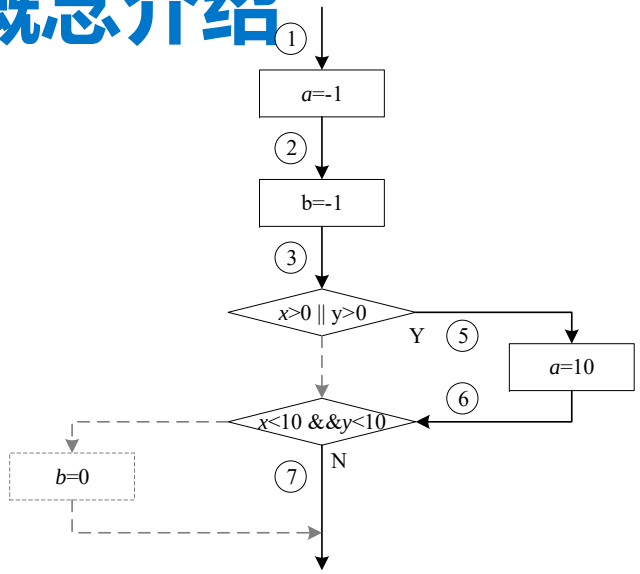


图 2(a) P1在t4=(20, 20)下的程序流程图

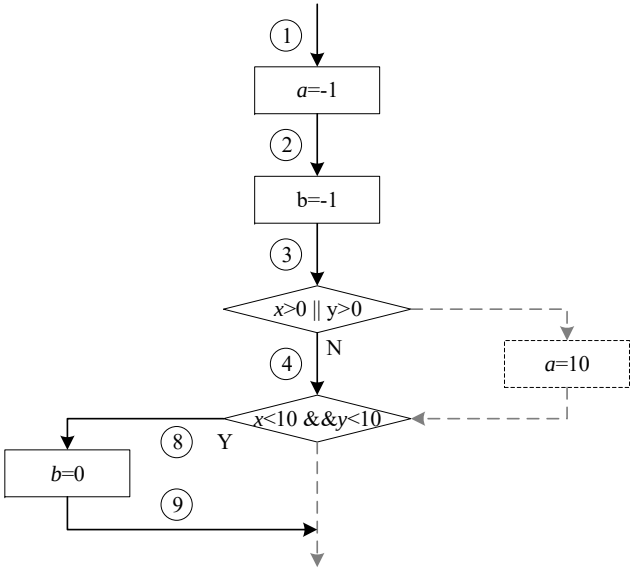


图 2(b) P1 在t5 =(-2, -2)下的程序流程图

表 1 P1在测试用例t_4-t_5下的分支覆盖表

测试用例	x	y	分支覆盖结果	
			x > 0 | | y > 0	x < 10 && y < 10
t_4	20	20	⑤	⑦
t_5	-2	-2	④	⑧

概念介绍

变异测试定义

- 变异测试也称为“变异分析”，是一种对测试数据集的有效性、充分性进行评估的技术，能为研发人员开展需求设计、单元测试、集成测试提供有效的帮助
- 变异测试通过对比源程序与变异程序在执行同一测试用例时差异来评价测试用例集的错误检测能力
- 在变异测试过程中，一般利用与源程序差异极小的简单变异体来模拟程序中可能存在的各种缺陷

概念介绍

变异测试应用场景

- 在软件测试时，若当前测试用例未能检测到软件缺陷，则存在两种情形：

  ① 软件已满足预设的需求，软件质量较高；

  ② 当前测试用例设计不够充分，不能有效检测到软件中的缺陷。

- 逻辑测试和路径测试方法，分别从程序实体覆盖和路径覆盖的角度来评估软件测试的充分性。然而，这些方法并不能直观地反映测试用例的缺陷检测能力

- 变异测试方法可用于度量测试用例缺陷检测能力

概念介绍

变异测试的基本假设

- 变异测试的可行性主要基于两点假设：
  - 熟练程序员假设，关注于熟练程序员的编程行为；
  - 变异耦合效应假设，关注于变异程序的缺陷类型。
- 熟练程序员假设是指熟练程序员由于开发经验丰富、编程水平较高，其编写的代码即使包含缺陷，也与正确代码十分接近。此时，针对缺陷代码仅需做微小的修改即可使代码恢复正确
- 变异耦合效应假设是指若测试用例能够杀死简单变异体，那么该测试用例也易于杀死复杂变异体

概念介绍

变异的定义

- 程序变异指基于预先定义的变异操作对程序进行修改，进而得到源程序变异程序（也称为变异体）的过程
  - 当源程序与变异程序存在执行差异时，则认为该测试用例检测到变异程序中的错误，变异程序被杀死
  - 当两个程序不存在执行差异时，则认为该测试用例没有检测到变异程序中的错误，变异程序存活

概念介绍

表 2 面向过程程序的变异算子

变异算子	描述
运算符变异	对关系运算符 “<”、“<=”、“>”、“>=” 进行替换，如将 “<” 替换为 “<=”
	对自增运算符 “++” 或自减运算符 “--” 进行替换，如将 “++” 替换为 “--”
	对与数值运算的二元算术运算符进行替换，如将 “+” 替换为 “-”
	将程序中的条件运算符替换为相反运算符，如将 “==” 替换为 “!=”
数值变异	对程序中整数类型、浮点数类型的变量取相反数，如将 “i” 替换为 “-i”
方法返回值变异	删除程序中返回值类型为void的方法
	对程序中方法的返回值进行修改，如将 “true” 修改为 “false”

概念介绍

表 3 面向对象程序的变异算子

变异算子	描述
继承变异	增加或删除子类中的重写变量
	增加、修改或重命名子类中的重写方法
	删除子类中的关键字super，如将 “return a*super.b” 修改为 “return a*b”
多态变异	将变量实例化为子类型
	将变量声明、形参类型改为父类型，如将 “Integer i” 修改为 “Object i”
	赋值时将使用变量替换为其它可用类型
重载变异	修改重载方法的内容，或删除重载方法
	修改方法参数的顺序或数量

概念介绍

变异测试过程

- 在变异测试过程中，程序与变异程序的执行差异主要表现为以下两个情形：
  - (1) 执行同一测试用例时，源程序和变异程序产生了不同的运行时状态
  - (2) 执行同一测试用例时，源程序和变异程序产生了不同的执行结果
- 根据满足执行差异要求的不同，可将变异测试分为弱变异测试 (Weak Mutation Testing) 和强变异测试 (Strong Mutation Testing)
- 在弱变异测试过程中，当情形 (1) 出现时就可认为变异程序被杀死，而在强变异测试过程中，只有情形 (1) 和 (2) 同时满足才可认为变异程序被杀死

概念介绍

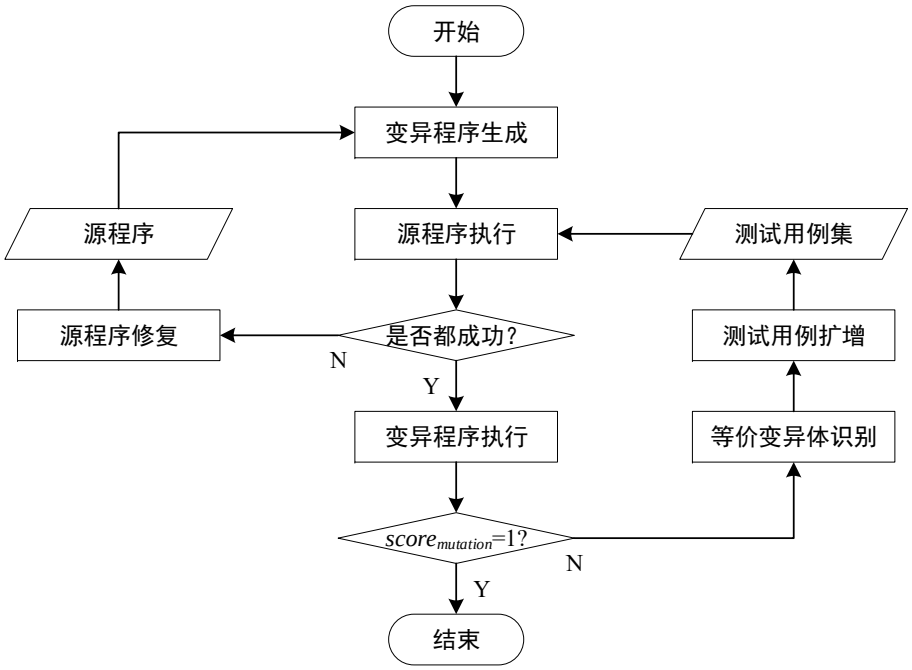


图 3 变异测试基本流程图

概念介绍

- 分支覆盖得分（Branch Score）是一种评价测试用例集充分检测有效性的度量指标
- 分支覆盖得分 $score_{\text{branch}}$ 的值介于 0 与 1 之间，数值越高，表明覆盖的分支数越多，测试用例集覆盖的流程越多，反之则越少
  - 当$score_{\text{branch}}$的值为 0 时，表明测试用例集没有覆盖任何一个分支
  - 当$score_{\text{branch}}$的值为 1 时，表明测试用例集覆盖了所有分支

$$score_{\text{branch}} = \frac{num_{\text{covered}}}{num_{total}}$$

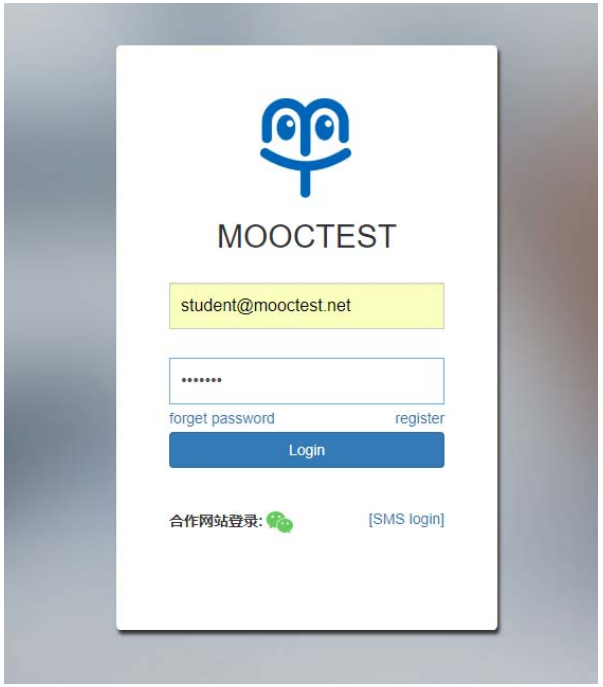
概念介绍

- 变异得分（Mutation Score）是一种评价测试用例集错误检测有效性的度量指标
- 变异得分 $score_{mutation}$ 的值介于 0 与 1 之间，数值越高，表明被杀死的变异程序越多，测试用例集的错误检测能力越强，反之则越低
  - 当$score_{mutation}$的值为 0 时，表明测试用例集没有杀死任何一个变异程序
  - 当$score_{mutation}$的值为 1 时，表明测试用例集杀死了所有非等价的变异程序

$$score_{mutation} = \frac{num_{killed}}{num_{total} - num_{equivalent}}$$

参赛流程

1.登录慕测官网(summer.mooctest.net)



参赛流程

2.点击报名

2018慕测9月双周赛

当前赛段：2018慕测9月双周赛1

晋级下一轮： 尚未获得下一轮资格

报名

2018"上海软件中心杯"开发者测试大赛

当前赛段：2018开发者测试秋季资格赛

晋级下一轮： 尚未获得下一轮资格

报名

2018"拓思杯"移动应用测试大赛

当前赛段：2018移动应用测试秋季资格赛

晋级下一轮： 尚未获得下一轮资格

报名

2018"拓思杯"移动应用测试大赛-团队赛

当前赛段：2018移动应用测试秋季团队资格赛

晋级下一轮： 尚未获得下一轮资格

报名

参赛流程

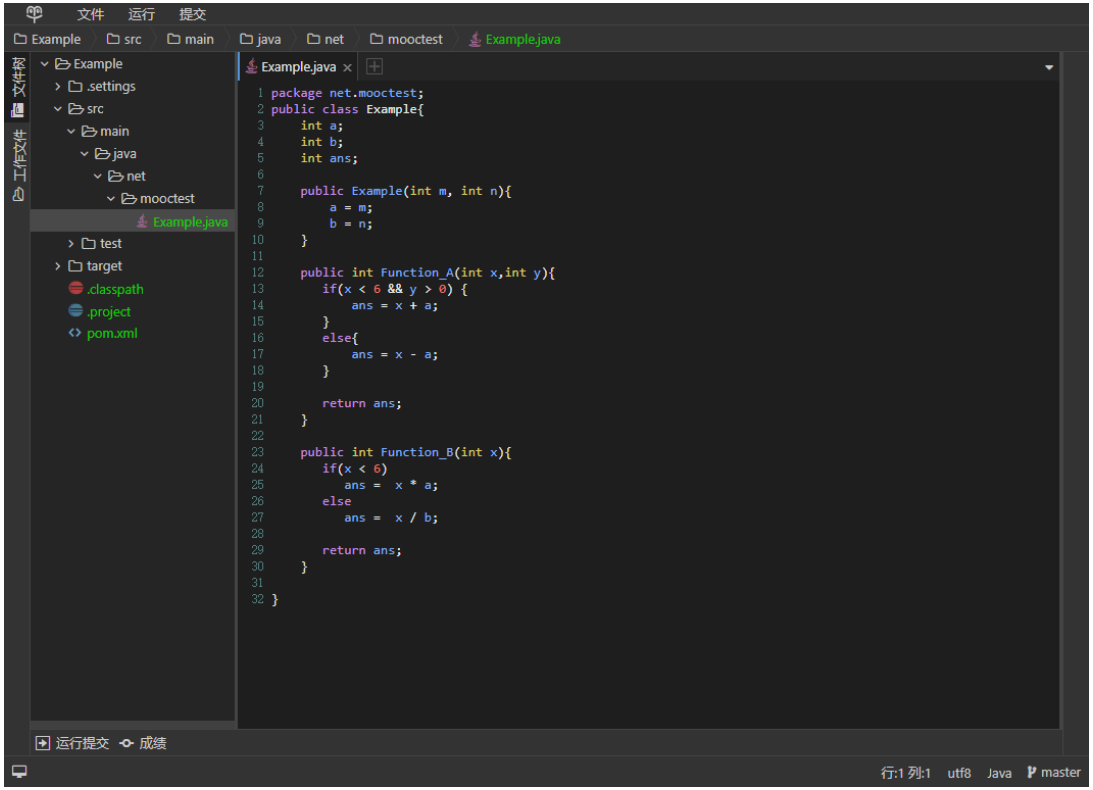
在线做题

参赛流程

3.点击在线做题

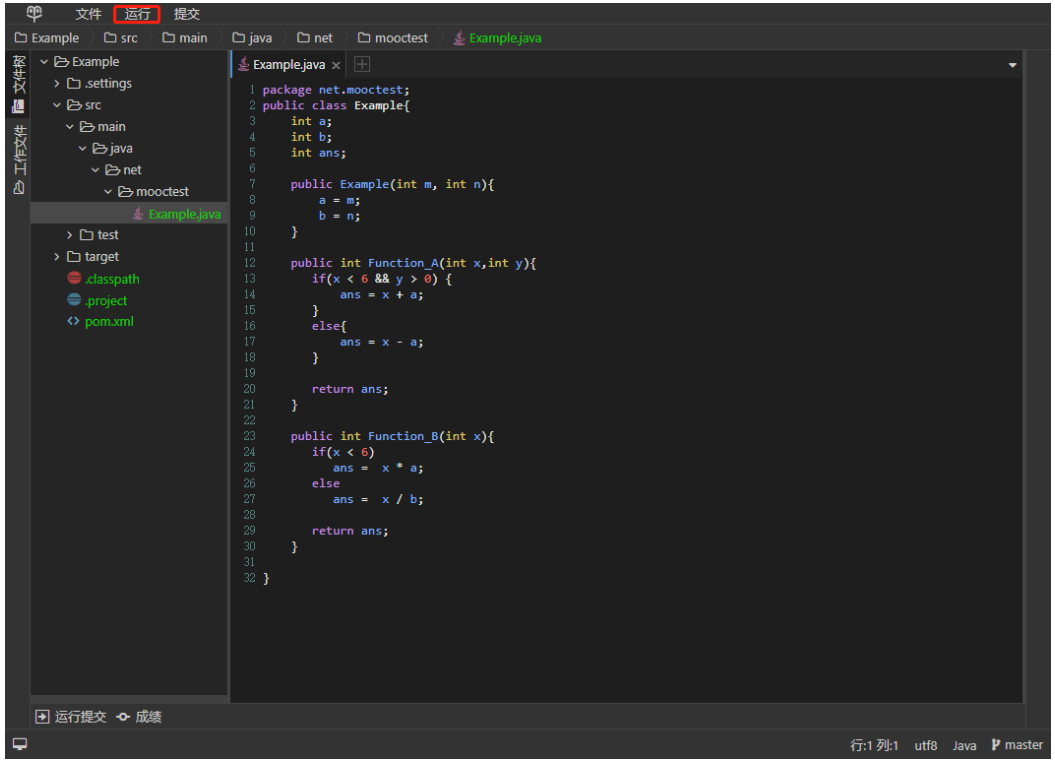
参赛流程

4.跳转到慕测提供的WEB IDE



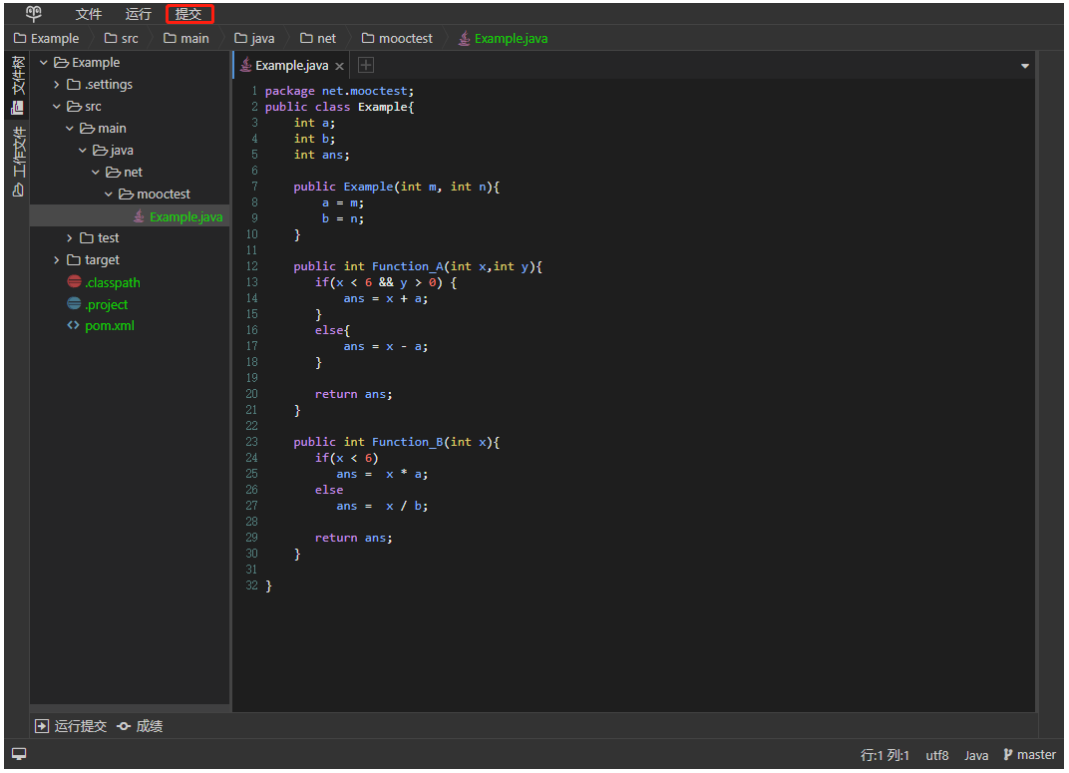
参赛流程

5.补充测试用例，点击运行本地运行



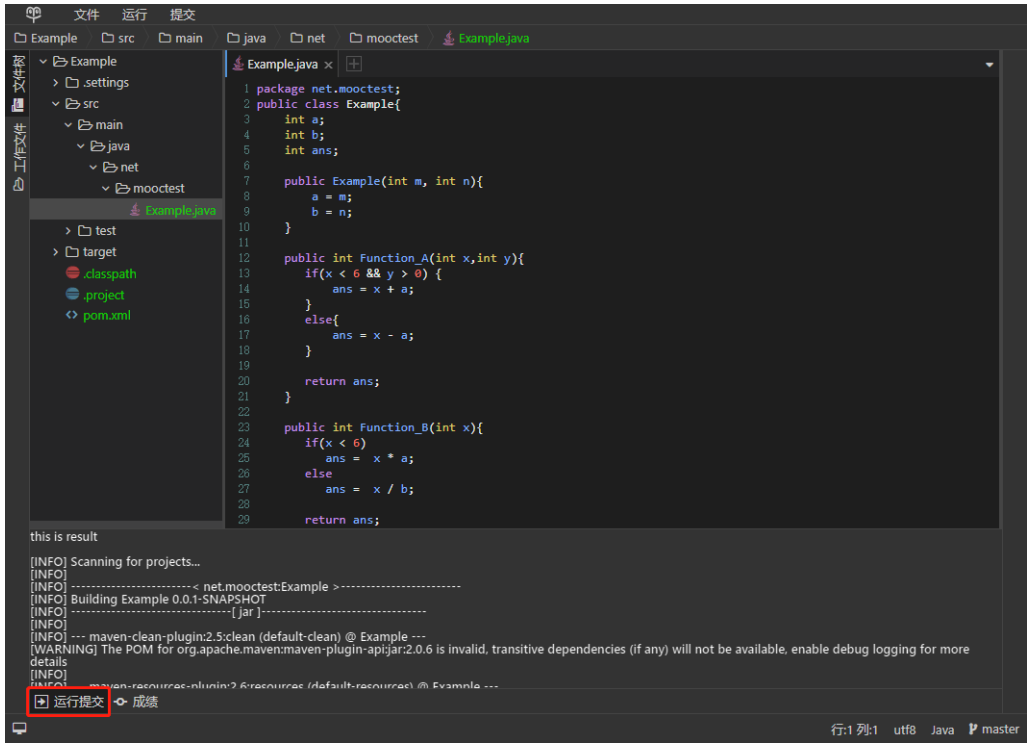
参赛流程

6.选择提交，提交到服务器



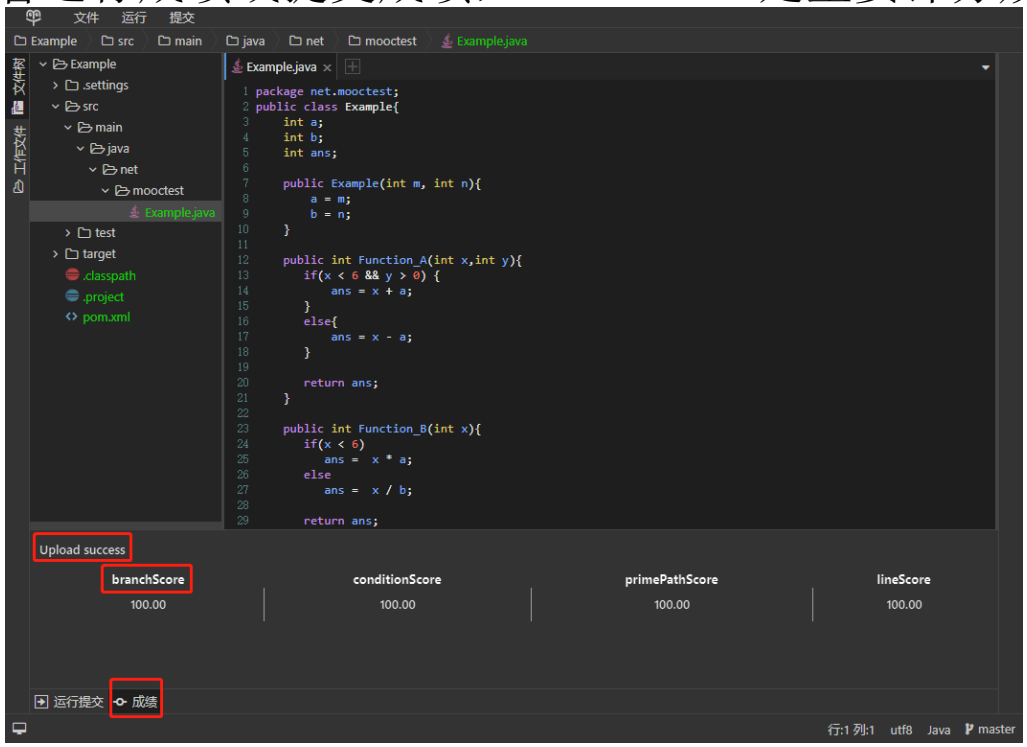
参赛流程

7.点击运行提交查看运行日志



参赛流程

8.点击成绩查看运行成绩或提交成绩，branchscore是主要评分规则



注：变异得分无法实时查看，使用选手最后一次的提交脚本统一运行，请确保最后一次提交的脚本是正确的

插件做题

参赛流程

1.登入Mooctest.net后，点击工具下载

参赛流程

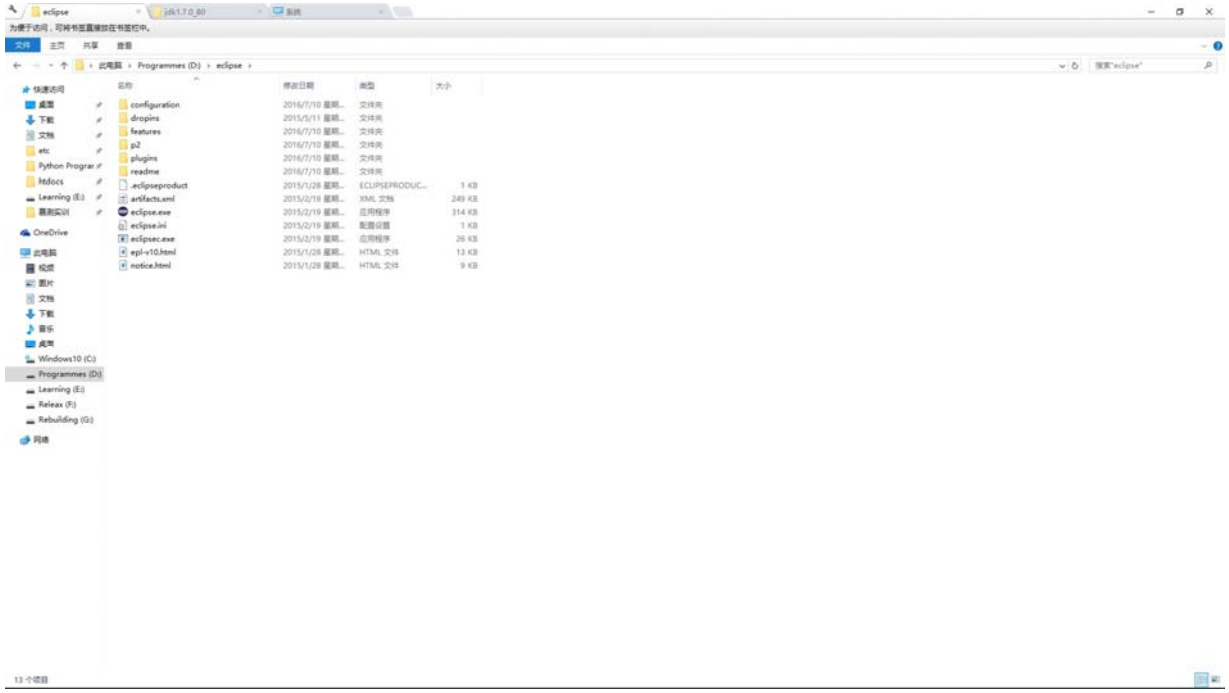
2.进入慕测插件下载页面

参赛流程

3.选择测试插件及Eclipse下载

参赛流程

4.解压下载的eclipse到安装目录(禁止中文路径)



参赛流程

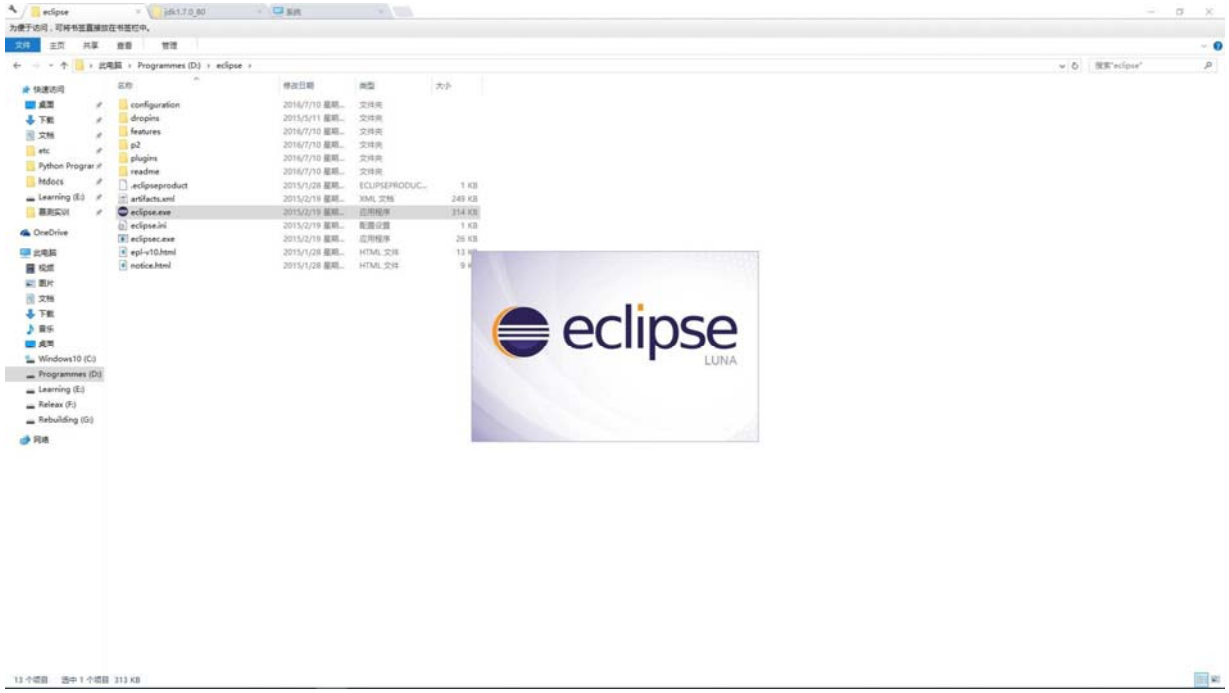
5.配置Java和Maven环境，并在本机命令行查看

```
Windows PowerShell
PS D:\> mvn -v
D:\
Apache Maven 3.5.0 (ff8f5e7444045639af65f6095c62210b5713f426; 2017-04-04T03:39:06+08:00)
Maven home: D:\Apache\Maven\bin\..
Java version: 1.8.0_131, vendor: Oracle Corporation
Java home: D:\Java\jdk1.8.0_131\jre
Default locale: zh_CN, platform encoding: GBK
OS name: "windows 10", version: "10.0", arch: "amd64", family: "windows"
PS D:\> java -version
java version "1.8.0_131"
Java(TM) SE Runtime Environment (build 1.8.0_131-b11)
Java HotSpot(TM) 64-Bit Server VM (build 25.131-b11, mixed mode)
PS D:\>
```

注：JDK版本和Maven版本没有具体要求，Maven版本超过3即可

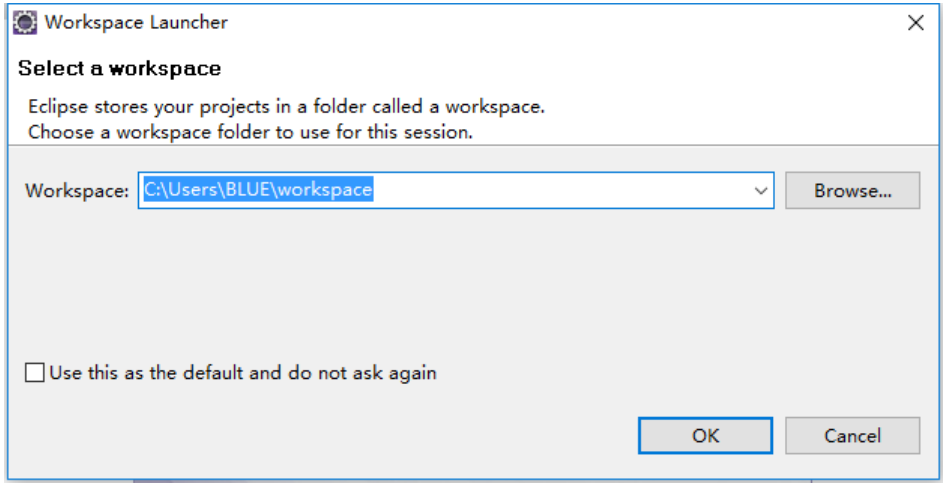
参赛流程

6.运行eclipse.exe



参赛流程

7.选择项目默认的新建目录



参赛流程

8.比赛页面点开题目详情

考试详情

考试名称	2018开发者测试秋季资格赛□ 进行中
起止时间	2018-09-20 12:00 - 2018-10-25 12:00
试卷名称	2018开发者测试秋季资格赛□试卷
出卷人	大赛组委会
难度	☆☆☆☆☆

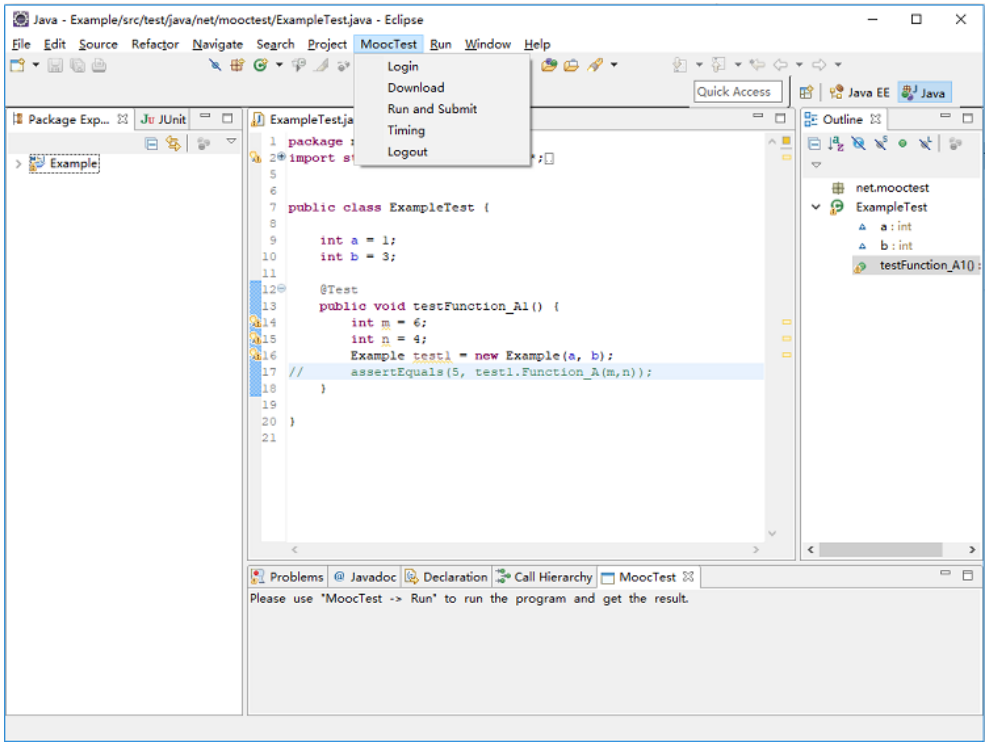
题目名称			分值占比	得分规则
1.MSD	已提交	在线做题	50%	branch 100%
暂无描述				
客户端秘钥	OaOGV8WM68DO70Fyr52SgQ==			
2.MatrixInverse	已提交	在线做题	50%	branch 100%

参赛流程

9.复制客户端密钥

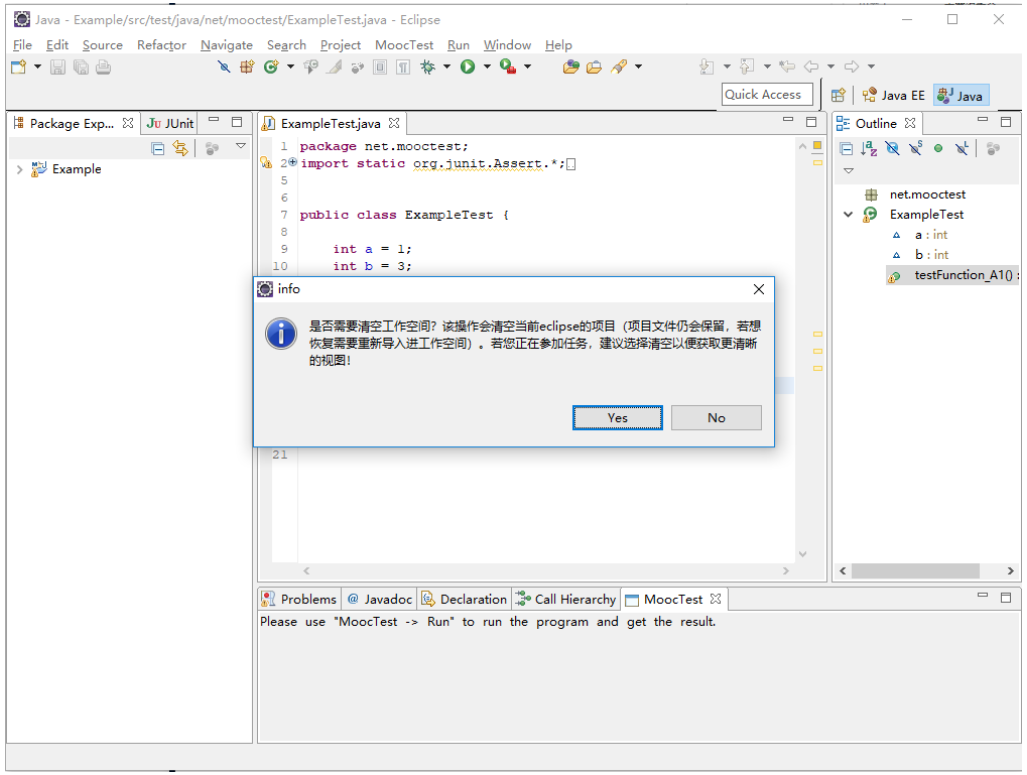
参赛流程

10.选择Mooctest-Login



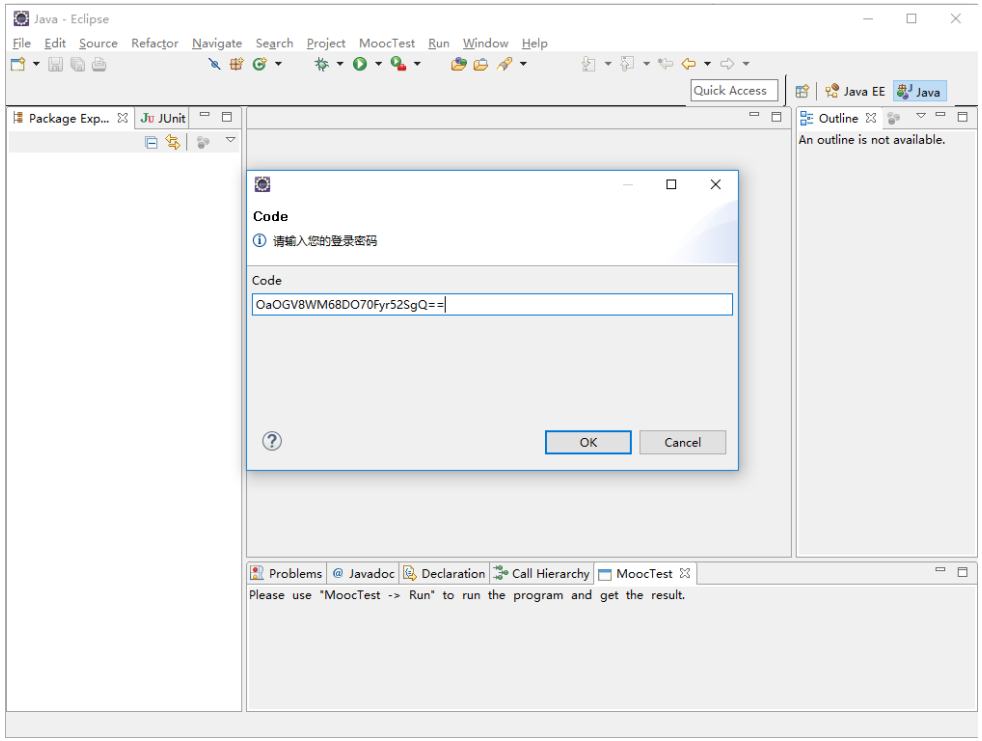
参赛流程

11.选择清空工作空间



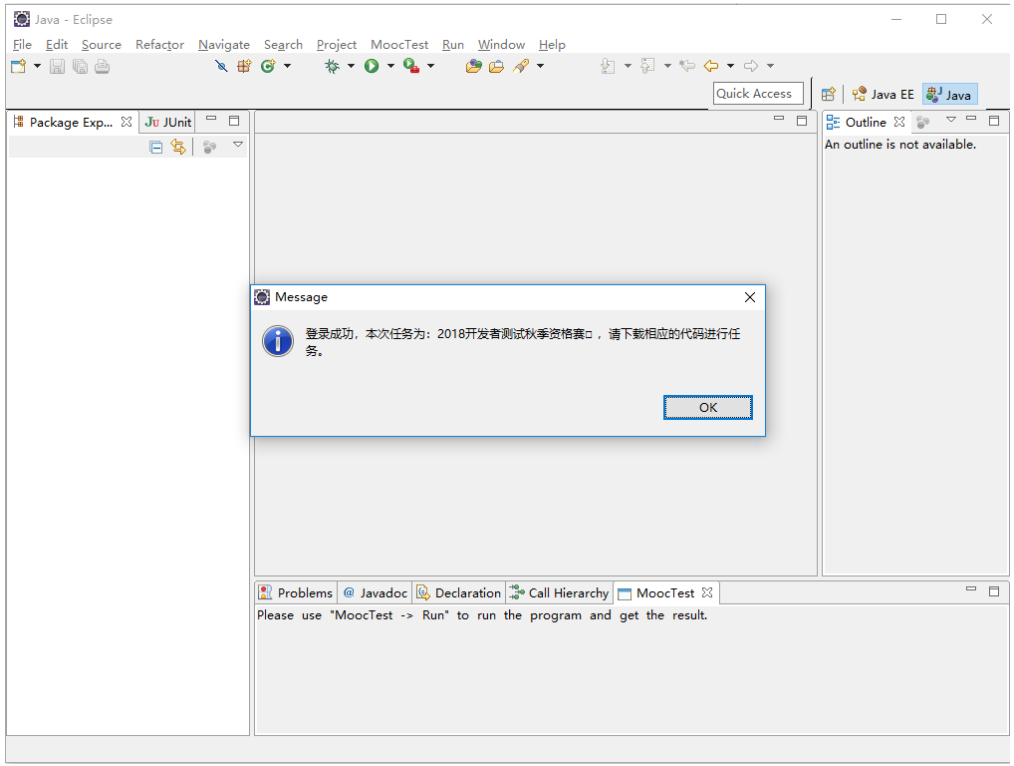
参赛流程

12.粘贴客户端密钥



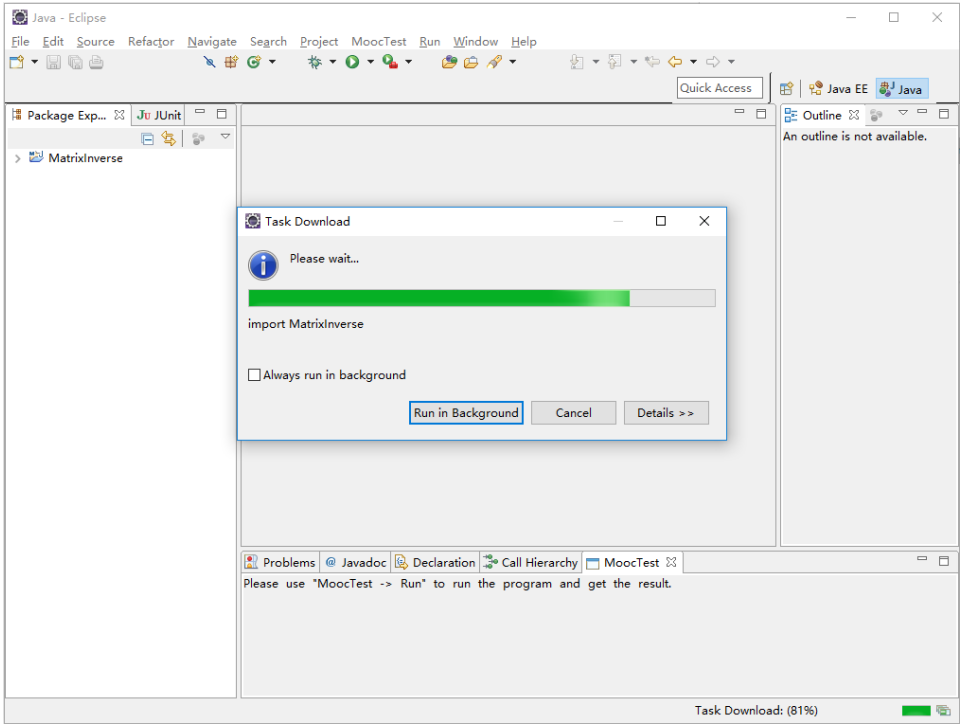
参赛流程

13.登录成功


Java - Eclipse
File Edit Source Refactor Navigate Search Project MoocTest Run Window Help
Quick Access
Java EE
Java
Package Exp...
JUnit
Outline
An outline is not available.
Message
登录成功，本次任务为：2018开发者测试秋季资格赛□，请下载相应的代码进行任务。
OK
Problems
Javadoc
Declaration
Call Hierarchy
MoocTest
Please use "MoocTest -> Run" to run the program and get the result.

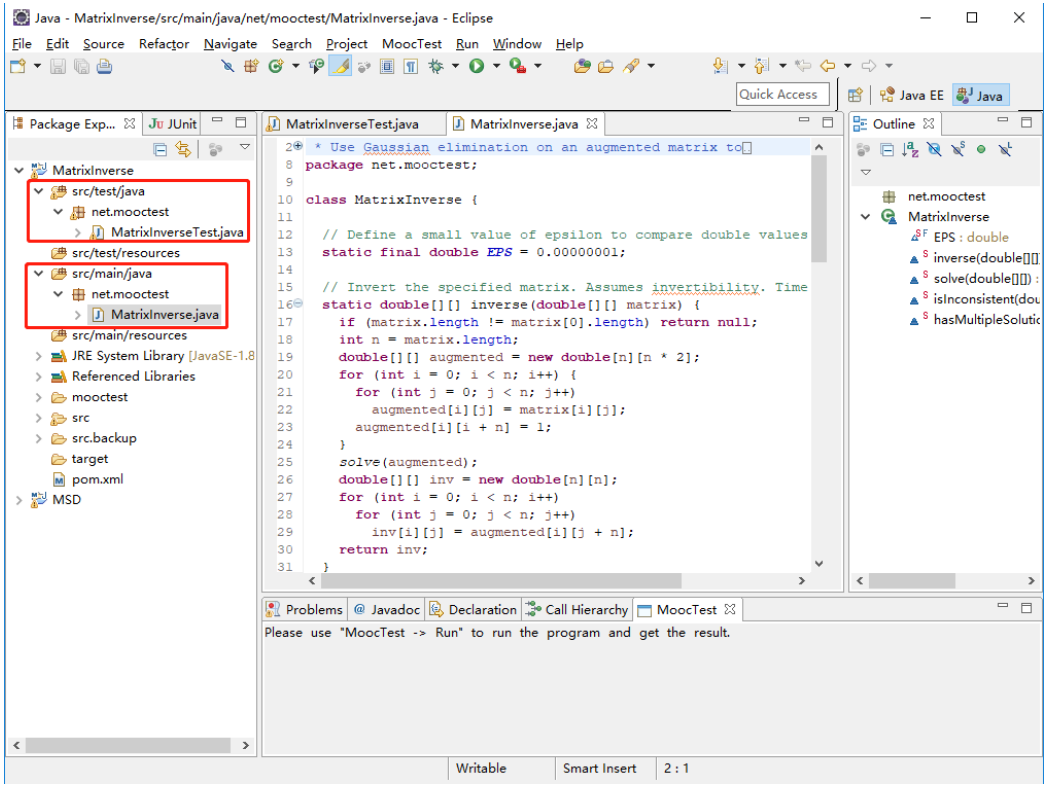
参赛流程

14.选择Mooctest-Download下载试题



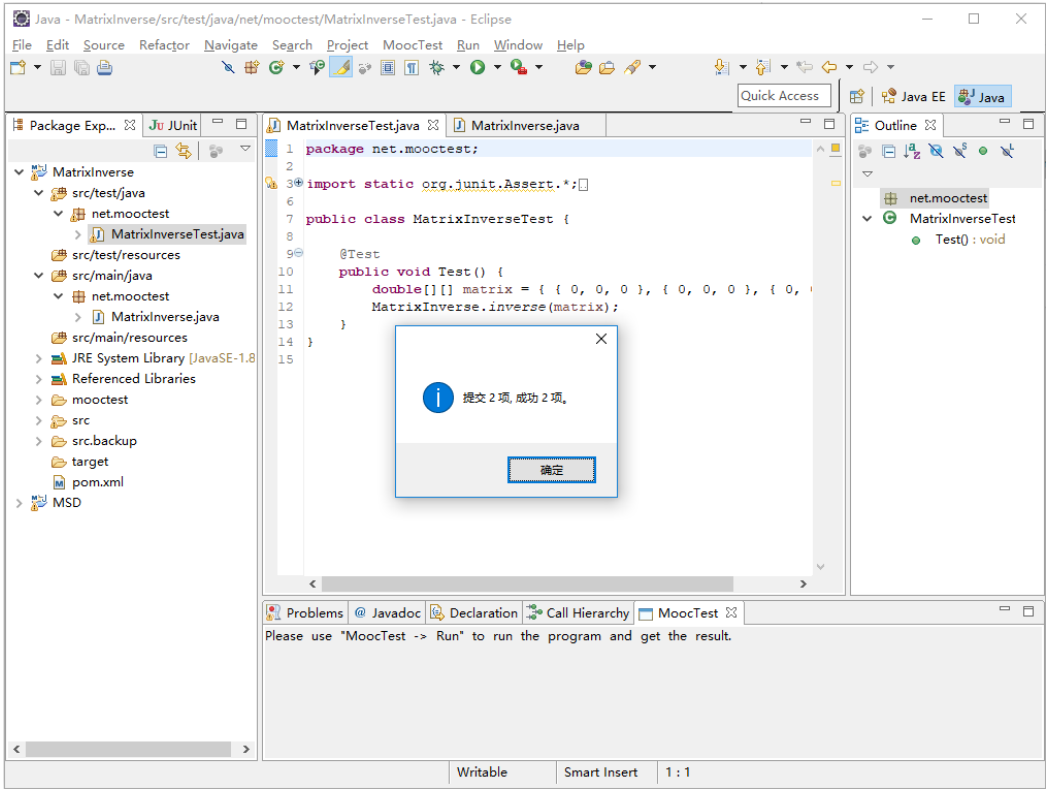
参赛流程

15.src/main/java和src/test/java中分别有源码和测试代码



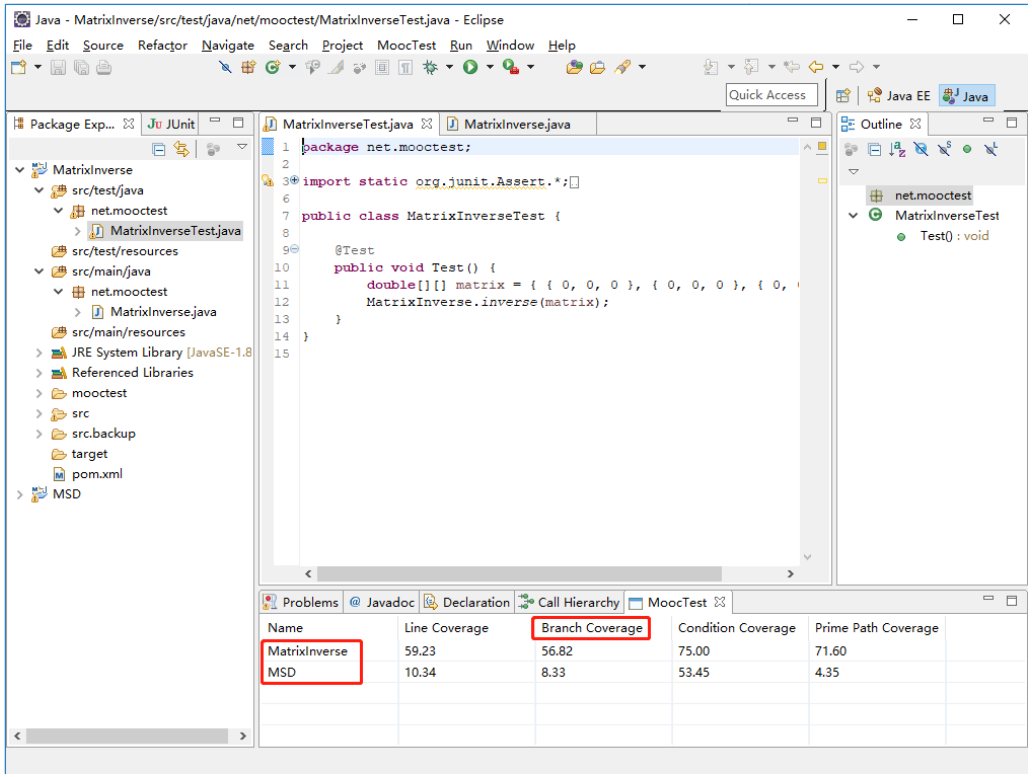
参赛流程

16.写好测试代码后，选择Mooctest-Run and Submit提交



参赛流程

17.可以看到每道题的单项得分，Branch为大赛得分点，插件端的成绩仅供参考最终成绩以主站为准



参赛流程

18.提交成功后，主站显示已提交

考试详情

考试名称	2018开发者测试秋季资格赛□ 进行中
起止时间	2018-09-20 12:00 - 2018-10-25 12:00
试卷名称	2018开发者测试秋季资格赛□试卷
出卷人	大赛组委会
难度	☆☆☆☆☆

题目名称		分值占比	得分规则
1.MSD 已提交	在线做题	50%	branch 100%
2.MatrixInverse 已提交	在线做题	50%	branch 100%

参赛流程

19.注意事项

1.本机安装有Eclipse的，可以下载插件进行单独安装

2.下载了含插件的Eclipse，就不需要再下载插件

3.Mac和Linux的mvn请关联到/usr/local/bin/mvn或者/usr/bin/env目录下

Maven配置方法

参赛流程

1.登录官网下载

系统要求

Java开发工具（JDK）	Maven 3.3需要JDK 1.7或以上执行-它仍然允许你建立对1.3和其他版本的JDK通过使用工具链
内存	没有最低要求
磁盘	大约10MB是Maven的安装要求。此外，额外的磁盘空间将 用于本地Maven仓库。你的本地存储库的大小将取决于使用不同但期望在 至少500MB。
操作系统	没有最低要求。启动脚本，包括shell脚本和Windows批处理文件。

Files

Maven is distributed in several formats for your convenience. Simply pick a ready-made binary distribution archive and follow the installation instructions. Use a source archive if you intend to build Maven yourself.

In order to guard against corrupted downloads/installations, it is highly recommended to verify the signature of the release bundles against the public KEYS used by the Apache Maven developers.

	Link	Checksum	Signature
Binary tar.gz archive	apache-maven-3.3.9-bin.tar.gz	apache-maven-3.3.9-bin.tar.gz.md5	apache-maven-3.3.9-bin.tar.gz.asc
Binary zip archive	apache-maven-3.3.9-bin.zip	apache-maven-3.3.9-bin.zip.md5	apache-maven-3.3.9-bin.zip.asc
Source tar.gz archive	apache-maven-3.3.9-src.tar.gz	apache-maven-3.3.9-src.tar.gz.md5	apache-maven-3.3.9-src.tar.gz.asc
Source zip archive	apache-maven-3.3.9-src.zip	apache-maven-3.3.9-src.zip.md5	apache-maven-3.3.9-src.zip.asc

windows下载apache-maven-3.3.9-bin.zip 版本

参赛流程

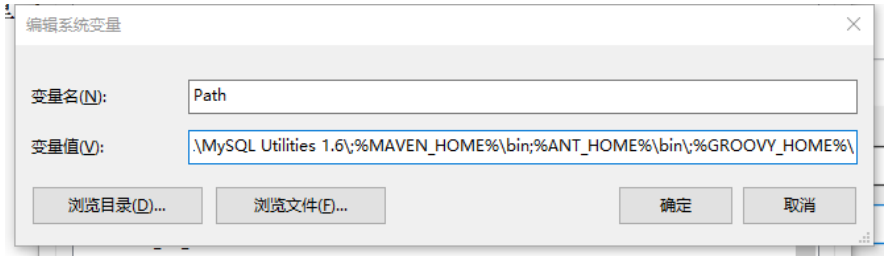
2.解压zip到本地

参赛流程

3.配置环境变量

右键“计算机”，选择“属性”，之后点击“高级系统设置”，点击“环境变量”，来设置环境变量，有以下系统变量需要配置：

新建系统变量 MAVEN_HOME 变量值：D:\Apache\Maven 编辑系统变量 Path 添加变量值：;%MAVEN_HOME%\bin



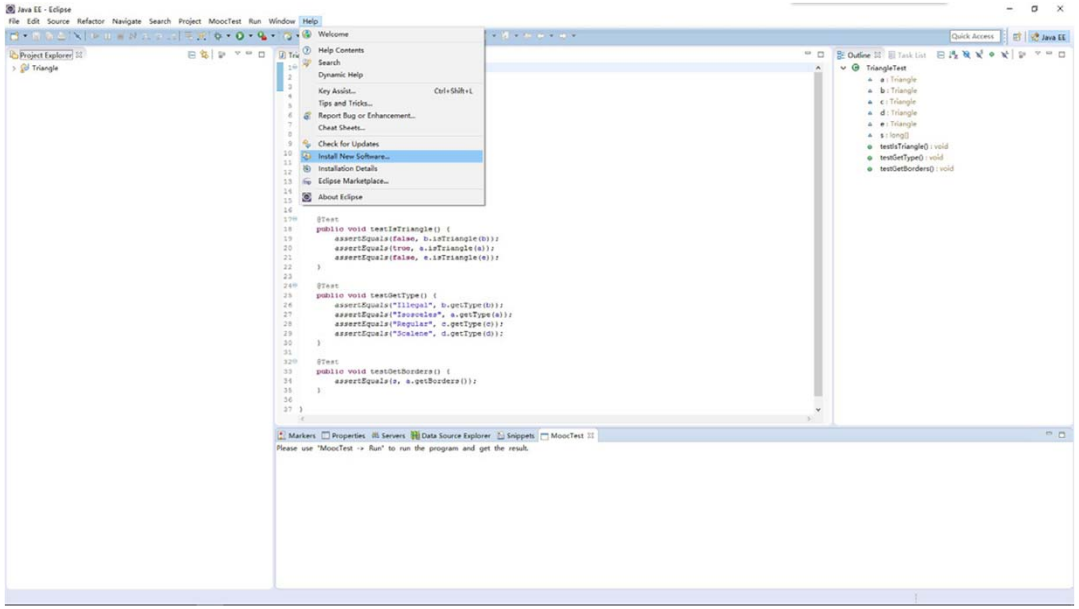
插件本地更新

参赛流程

1.进入慕测下载页，选中慕测插件下载

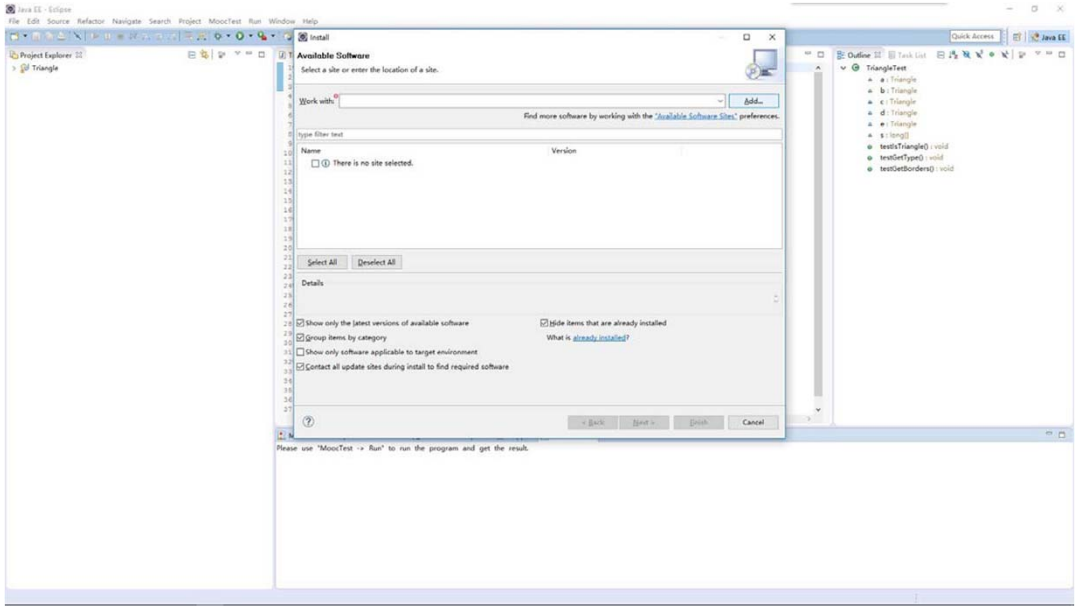
参赛流程

2.打开Eclipse，Help-Install New Software



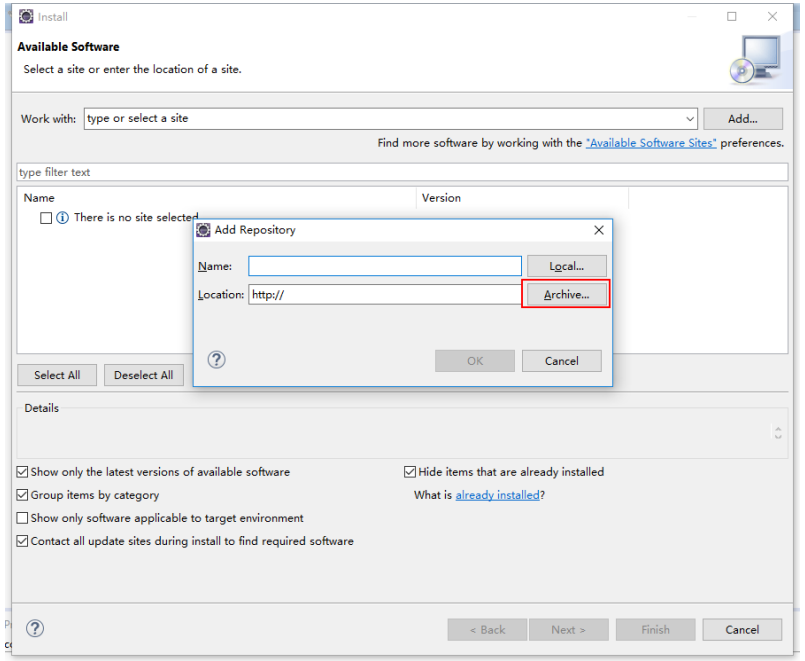
参赛流程

3.点击Add


Install
Available Software
Select a site or enter the location of a site.
Work with:
Add...
Find more software by working with the "Available Software Sites" preferences.
Name
Version
There is no site selected.
Select All
Deselect All
Details
Show only the latest versions of available software
Hide items that are already installed
Group items by category
What is already installed?
Show only software applicable to target environment
Contact all update sites during install to find required software
< Back
Next >
Finish
Cancel

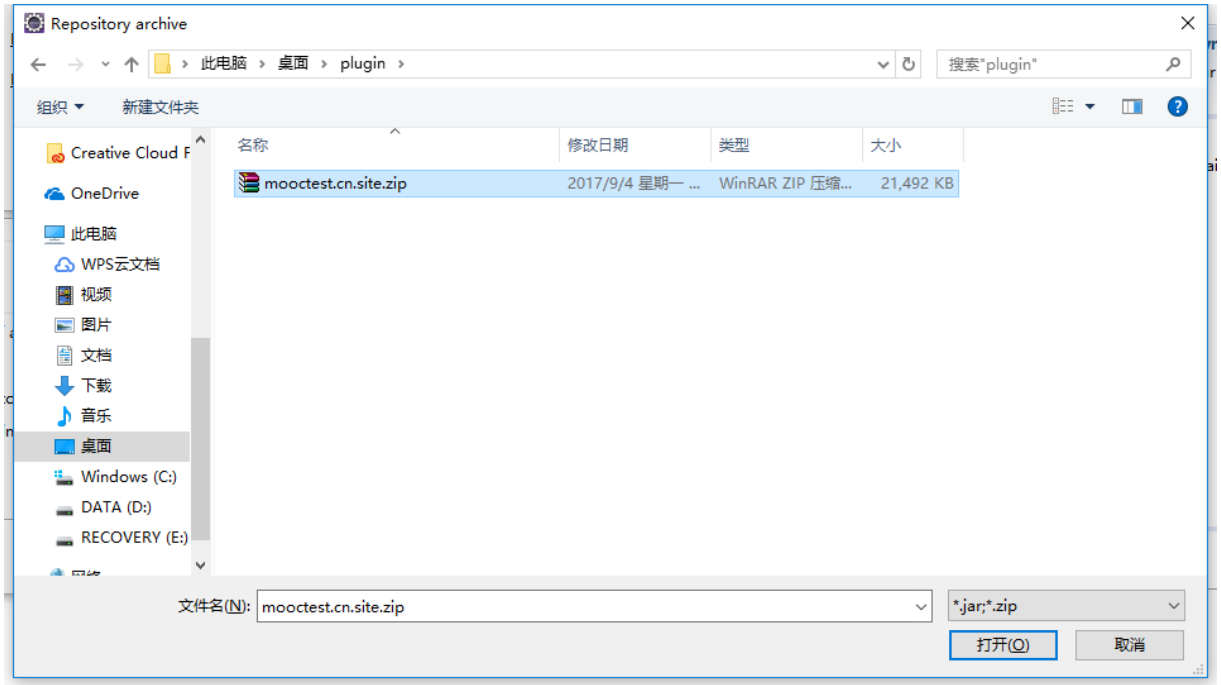
参赛流程

4.选择Archive



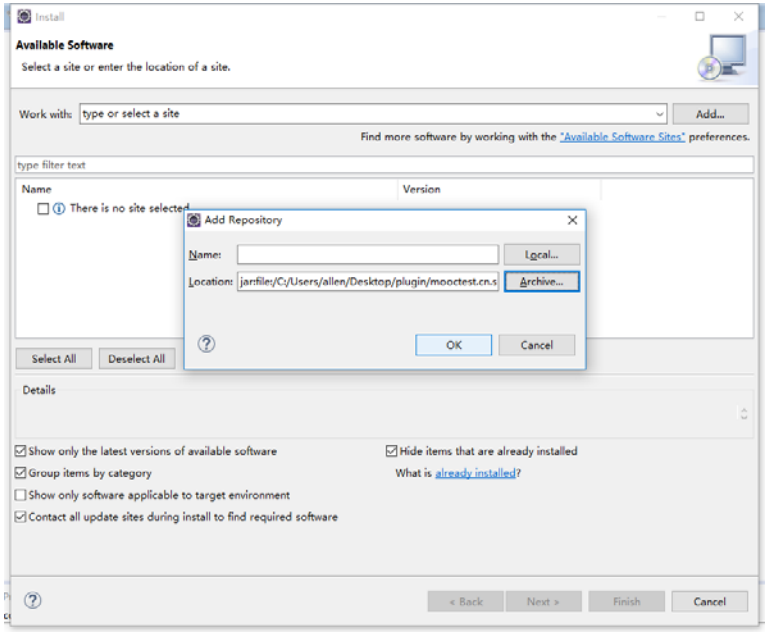
参赛流程

5.选择下载的插件压缩包



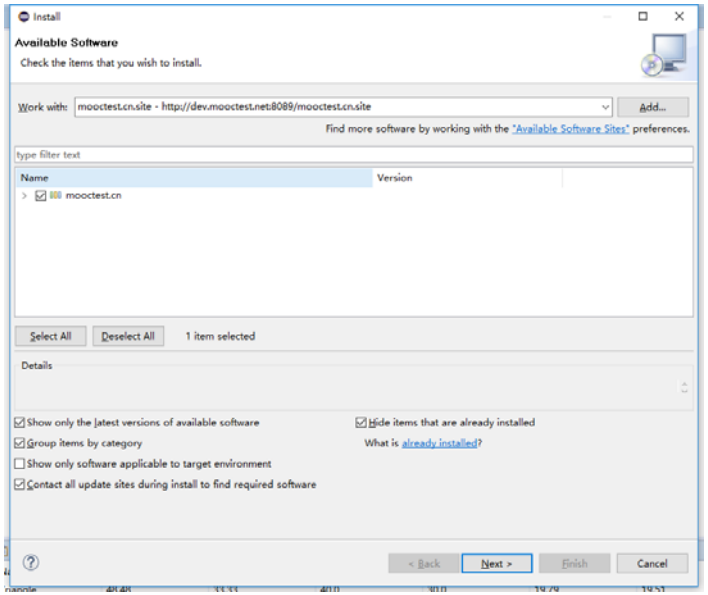
参赛流程

6.点击OK


Install
Available Software
Select a site or enter the location of a site.
Work with: type or select a site
Add...
Find more software by working with the "Available Software Sites" preferences.
type filter text
Name
Version
There is no site selected.
Add Repository
Name:
Local...
Location: jar:file:/C:/Users/allen/Desktop/plugin/mooctest.cn.s
Archive...
OK
Cancel
Select All
Deselect All
Details
Show only the latest versions of available software
Group items by category
Show only software applicable to target environment
Contact all update sites during install to find required software
Hide items that are already installed
What is already installed?
< Back
Next >
Finish
Cancel

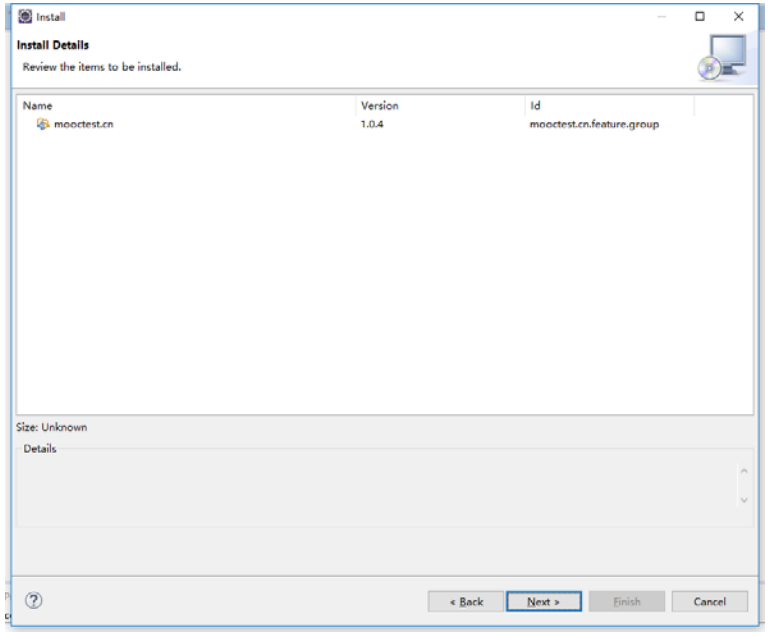
参赛流程

7.选中慕测插件，点击Next



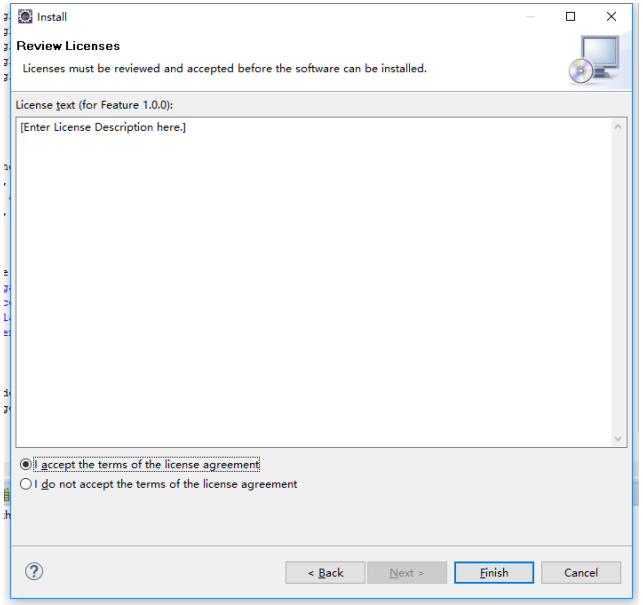
参赛流程

8.Next


Install
Install Details
Review the items to be installed.
Name
Version
Id
mooctest.cn
1.0.4
mooctest.cn.feature.group
Size: Unknown
Details
< Back
Next >
Finish
Cancel

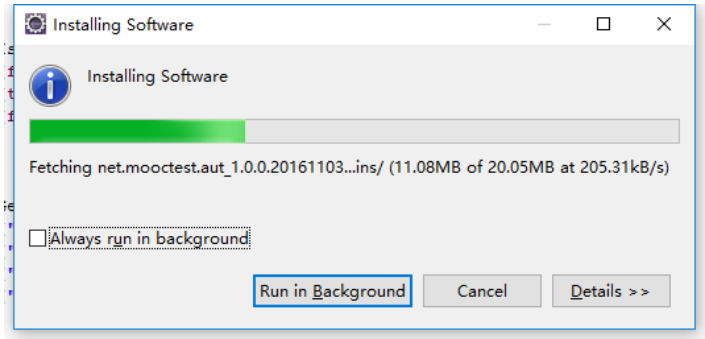
参赛流程

9.选择接受，点击Finish



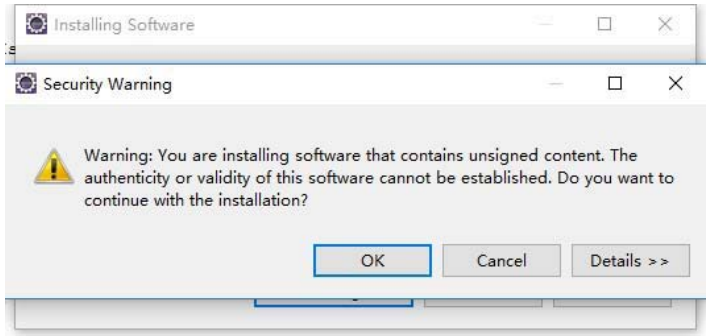
参赛流程

10.安装过程截图



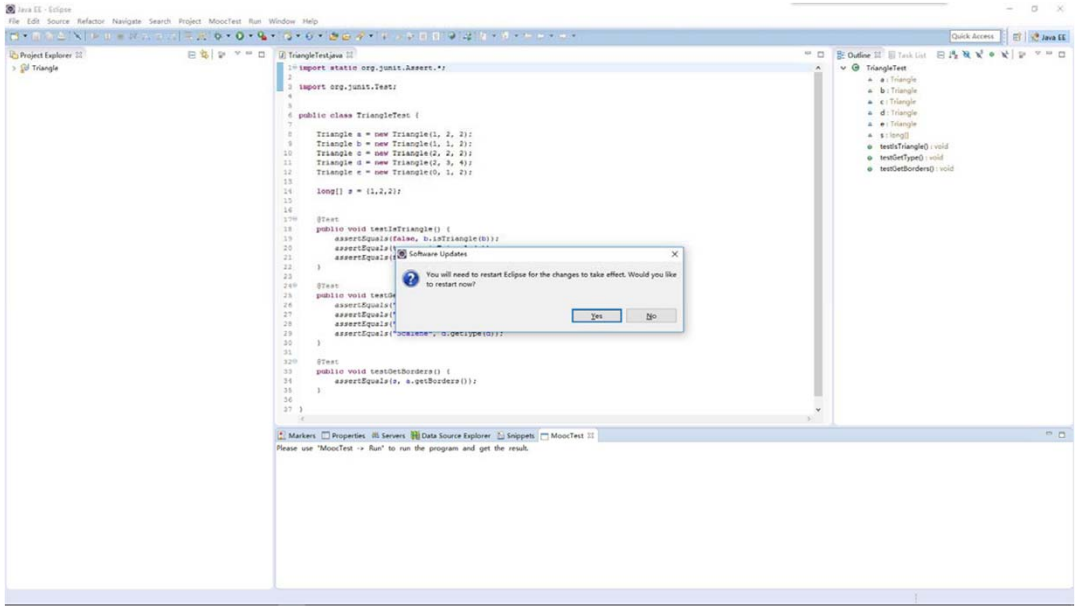
参赛流程

11.选择OK


Installing Software
Security Warning
Warning: You are installing software that contains unsigned content. The authenticity or validity of this software cannot be established. Do you want to continue with the installation?
OK
Cancel
Details >>

参赛流程

12.选择Yes， 重启eclipse



编程建议

慕课资料

https://www.coursera.org/learn/ruanjian-ceshi

https://www.imooc.com/learn/356

编程建议

必须使用断言

确认最后提交的测试用例全部正确，Webide或者客户端能运行得分

使用setup()和teardown()方法进行环境初始化

使用timeout限制测试用例运行时间，避免测试用例出现无响应情况

Evosuite生成的捕获异常的自动测试用例，可能导致变异无法跑通，学生自己注意

编程建议

对double类型进行判断

```
@Test
public void test() {
    double a = 1.0;
    assertEquals(1.0,a);
}
}
```

The method assertEquals(double, double) from the type Assert is deprecated

2 quick fixes available:

- Add @SuppressWarnings 'deprecation' to 'test()'
- Configure problem severity

Press 'F2' for focus

```
@Test
public void test() {
    double a = 1.0;
    assertEquals(1.0,a,0.00001);
}
```

void org.junit.Assert.assertEquals(double expected, double actual, double delta)

Asserts that two doubles are equal to within a positive delta. If they are not, an `AssertionError` is thrown. If the expected value is infinity then the delta value is ignored.NaNs are considered equal: `assertEquals(Double.NaN, Double.NaN, *)` passes

Parameters:

expected expected value
actual the value to check against expected
delta the maximum delta between expected and actual for which both numbers are still considered equal.

Press 'F2' for focus

编程建议

对控制台输出进行测试

```java
import static org.junit.Assert.*;

public class BinPackageTest {

    String sep;
    BinPackage bp;
    PrintStream console = null;
    ByteArrayInputStream in = null;
    ByteArrayOutputStream out = null;
    InputStream input = null;

    @Before
    public void setUp() throws Exception {
        bp = new BinPackage();
        out = new ByteArrayOutputStream();
        input = System.in;
        console = System.out;
        System.setOut(new PrintStream(out));
        sep = System.getProperty("line.separator");
    }

    @After
    public void tearDown() throws Exception {
        System.setIn(input);
        out.close();
        System.setOut(console);
    }

    @Test(timeout=4000)
    public void test1() {
        in = new ByteArrayInputStream(("2" + sep + "1 2" + sep +"3" + sep + "2 1 1" + sep + "2 2 2" + sep + "2 3 2").getBytes());
        System.setIn(in);
        bp.entrance();
        String ans = out.toString();
        assertEquals("false" + sep + "true" + sep + "2 2" + sep + "false" + sep, ans);
    }
```

编程建议

对异常报错进行测试

```java
import static org.junit.Assert.*;

import org.junit.Rule;
import org.junit.Test;
import org.junit.rules.ExpectedException;

public class AccountTest {
    private static final double DOUBLE_DELTA = 1e-15;

    @Rule
    public ExpectedException thrown= ExpectedException.none();

    @Test(timeout=4000)
    public void test1() {
        Bank bank = new Bank();
        Account checkingAccount = new Account(Account.CHECKING);
        Customer bill = new Customer("Bill").openAccount(checkingAccount);
        bank.addCustomer(bill);
        thrown.expect(IllegalArgumentException.class);
        thrown.expectMessage("amount must be greater than zero");
        checkingAccount.withdraw(-100.0);
    }
```

软件定义世界

质量保障未来

MOOCTEST.org