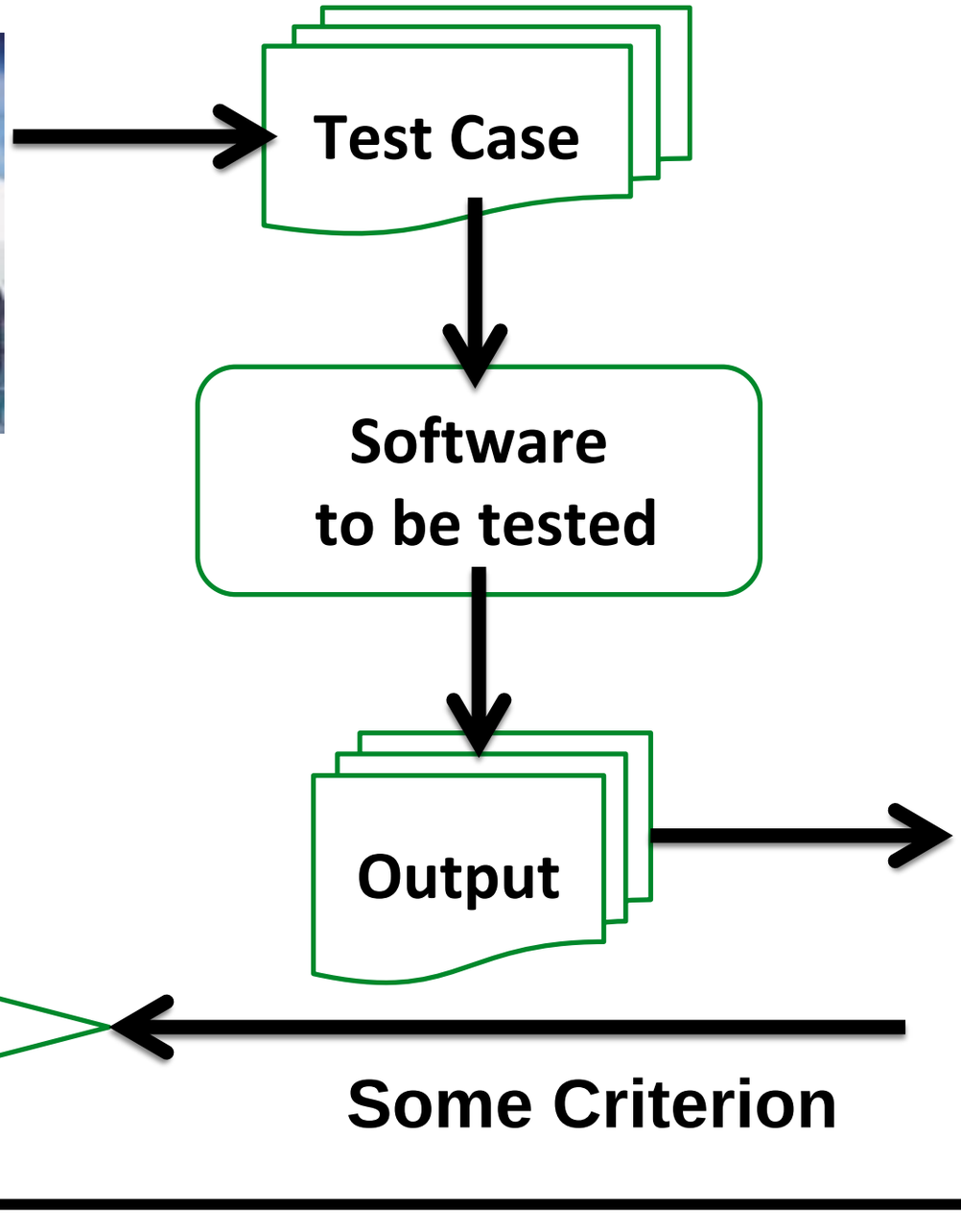
SENG3320/6320:
Software Verification and Validation

Software Testing Tools – JUnit

A Typical Software Testing Process

Test case generation



```
package test;

public class myClass {

    public static void main(String[] args) {
        myClass t = new myClass();
        int n = t.checkbool(10, true, false);
        System.out.println("the check boolean result is " + n);

          n = t.checkbool(11, false, false);
        System.out.println("the check boolean result is " + n);

          n = t.checkbool(10, true, true);
        System.out.println("the check boolean result is " + n);
}
   public int checkbool(int x, boolean a, boolean b) {
        if(a)
              x++;
        if(b)
              x--;
        return x;
    }
}
```

```
the check boolean result is 11
the check boolean result is 11
the check boolean result is 10
```

Tool support - testing framework

- xUnit
  - Created by Kent Beck in 1989
    - This is the same guy who invented XP and TDD
    - The first one was sUnit (for smalltalk)
    - There are about 70 xUnit frameworks for corresponding languages
- JUnit
  - A simple, flexible, easy-to-use, open-source, and practical xUnit framework for Java.
  - Can deal with a large and extensive set of test cases.
  - Refer to www.junit.org.

Terms

- Test Case: a set of test inputs, execution conditions, and expected results developed for a particular objective, such as to exercise a particular program path or to verify compliance with a specific requirement
- Test oracle: The expected outputs of software for given input. It is a part of a test case.
- Test driver: a software framework that can load a collection of test cases or a test suite.
- Test suite: a collection of test cases.

```
package test;


import static org.junit.jupiter.api.Assertions.*;
import org.junit.jupiter.api.Test;

public class junit_test {
myClass t;


@Test
void testBool_1() {
t = new myClass();
assertEquals(11, t.checkbool(10, true, false));
assertEquals(10, t.checkbool(10, true, true));
}

}
```

Test case

Test oracle

assertEquals([msg], expected, actual)	Check that the two values are equal

Limitation: Not clear which test case caused the failure

```
package test;


import static org.junit.jupiter.api.Assertions.*;
import org.junit.jupiter.api.Test;

public class junit_test {
myClass t;


@Test
void testBool_1() {
t = new myClass();
assertEquals(11, t.checkbool(10, true, false));
}

@Test
void testBool_2() {
t = new myClass();
assertEquals(10, t.checkbool(10, true, true));
}

}
```

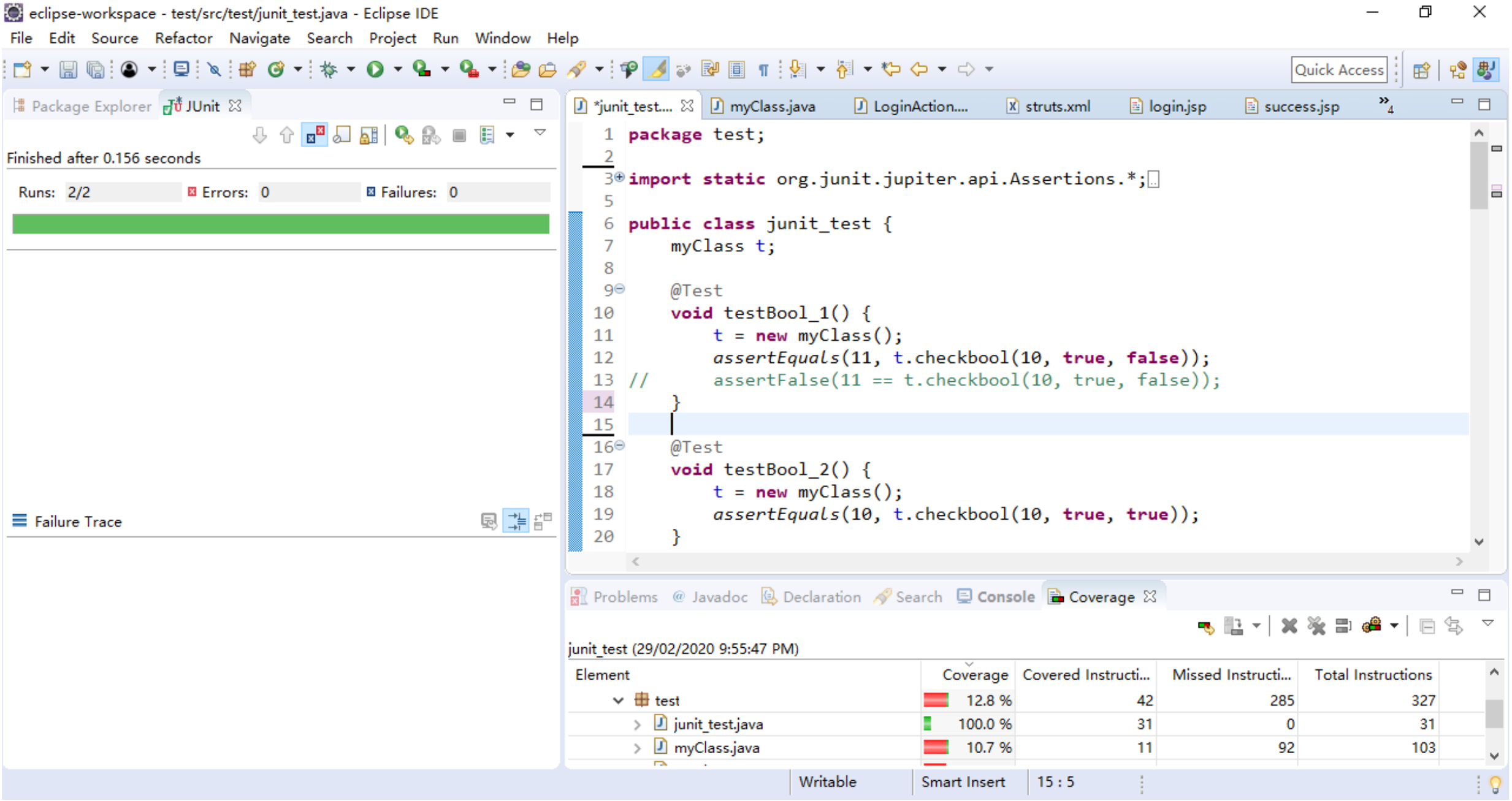
Test case

Test oracle

A Better Way

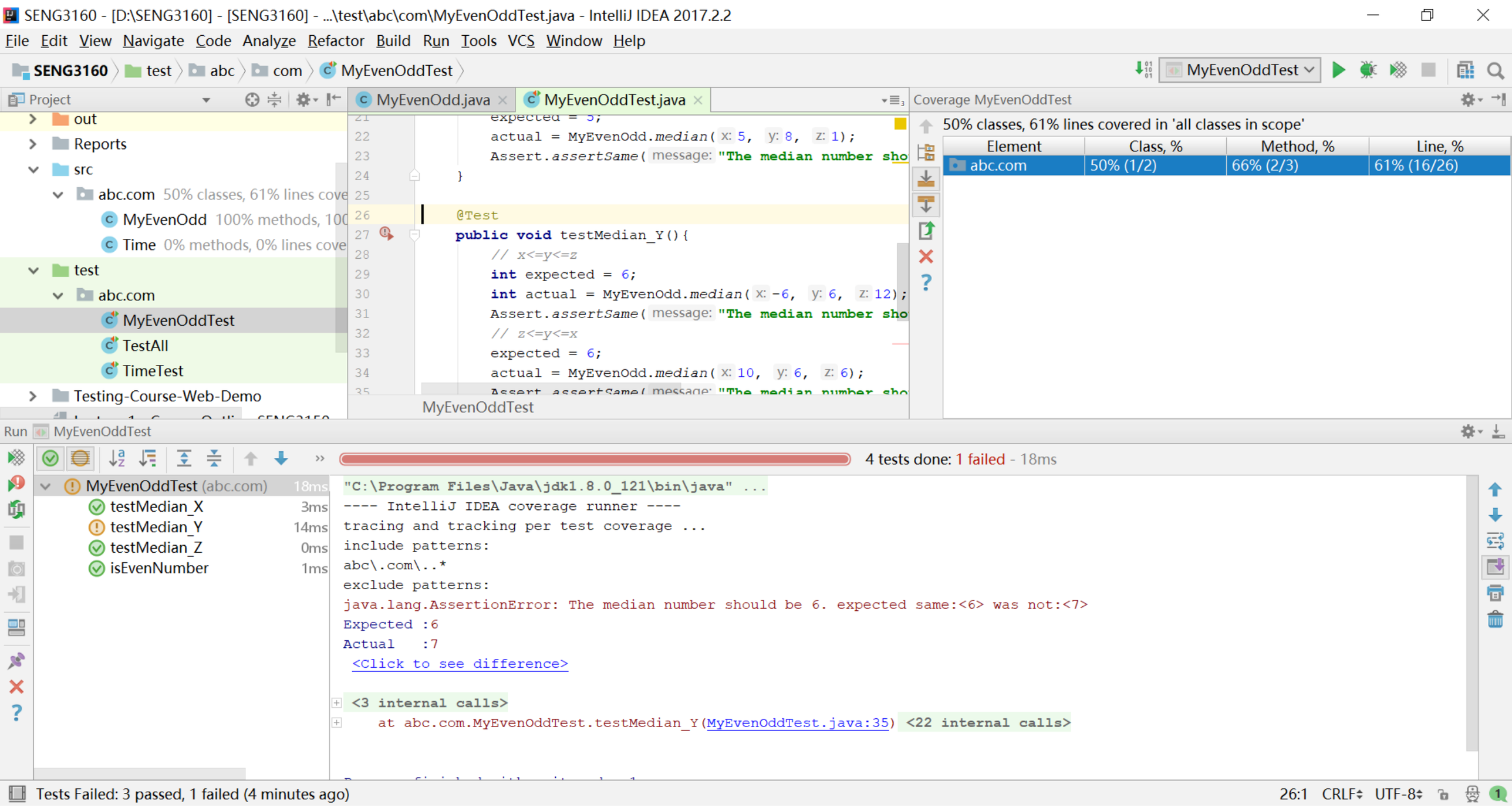
Eclipse Support

- Junit is supported by Eclipse



IntelliJ Support

- Junit is also supported by others IDEs such as IntelliJ



[Prepare for testing—IntelliJ IDEA (jetbrains.com)]

JUnit: annotations

Annotation	Description
@Test	Identify test methods
@Test (timeout=100)	Fail if the test takes more than 100ms
@Before	Execute before each test method
@After	Execute after each test method
@BeforeClass	Execute before each test class
@AfterClass	Execute after each test class
@Ignore	Ignore the test method

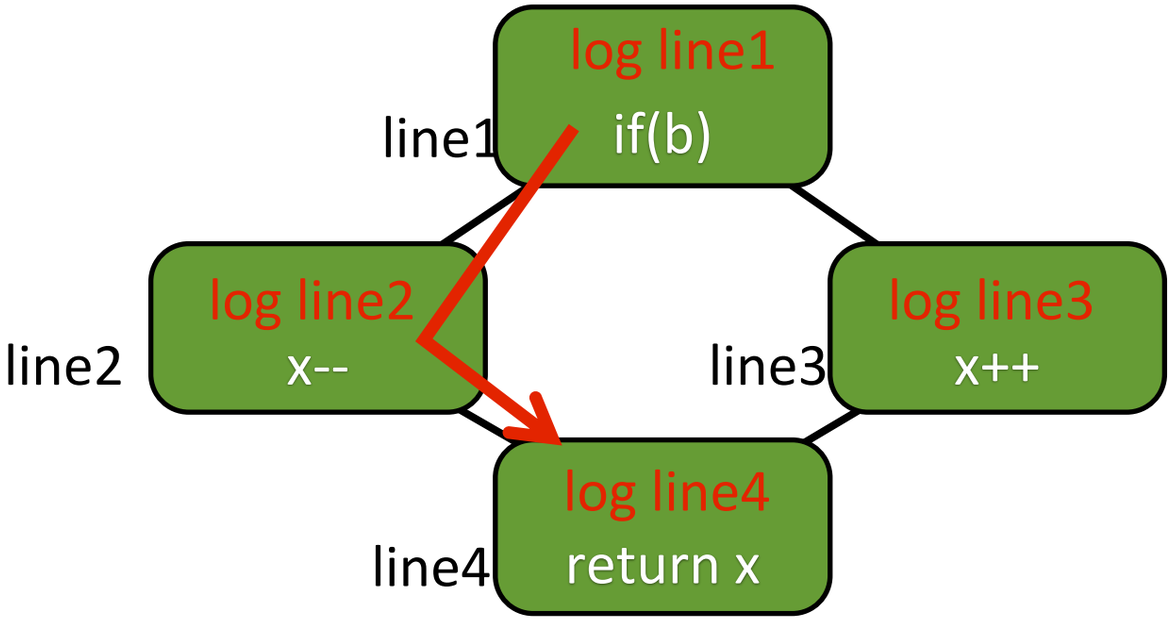
JUnit: assertions

Assertion	Description
fail([msg])	Let the test method fail, optional msg
assertTrue([msg], bool)	Check that the boolean condition is true
assertFalse([msg], bool)	Check that the boolean condition is false
assertEquals([msg], expected, actual)	Check that the two values are equal
assertNull([msg], obj)	Check that the object is null
assertNotNull([msg], obj)	Check that the object is not null
assertSame([msg], expected, actual)	Check that both variables refer to the same object
assertNotSame([msg], expected, actual)	Check that variables refer to different objects

Software Testing Tools – EclEmma

Coverage collection: mechanism

- The code under test is instrumented (source/binary)
  - Log code that writes to a trace file is inserted in every branch, statement etc.
- When the instrumented code is executed, the coverage info will be written to trace file



Coverage file

line1
line2
line4

Tool support - coverage collection

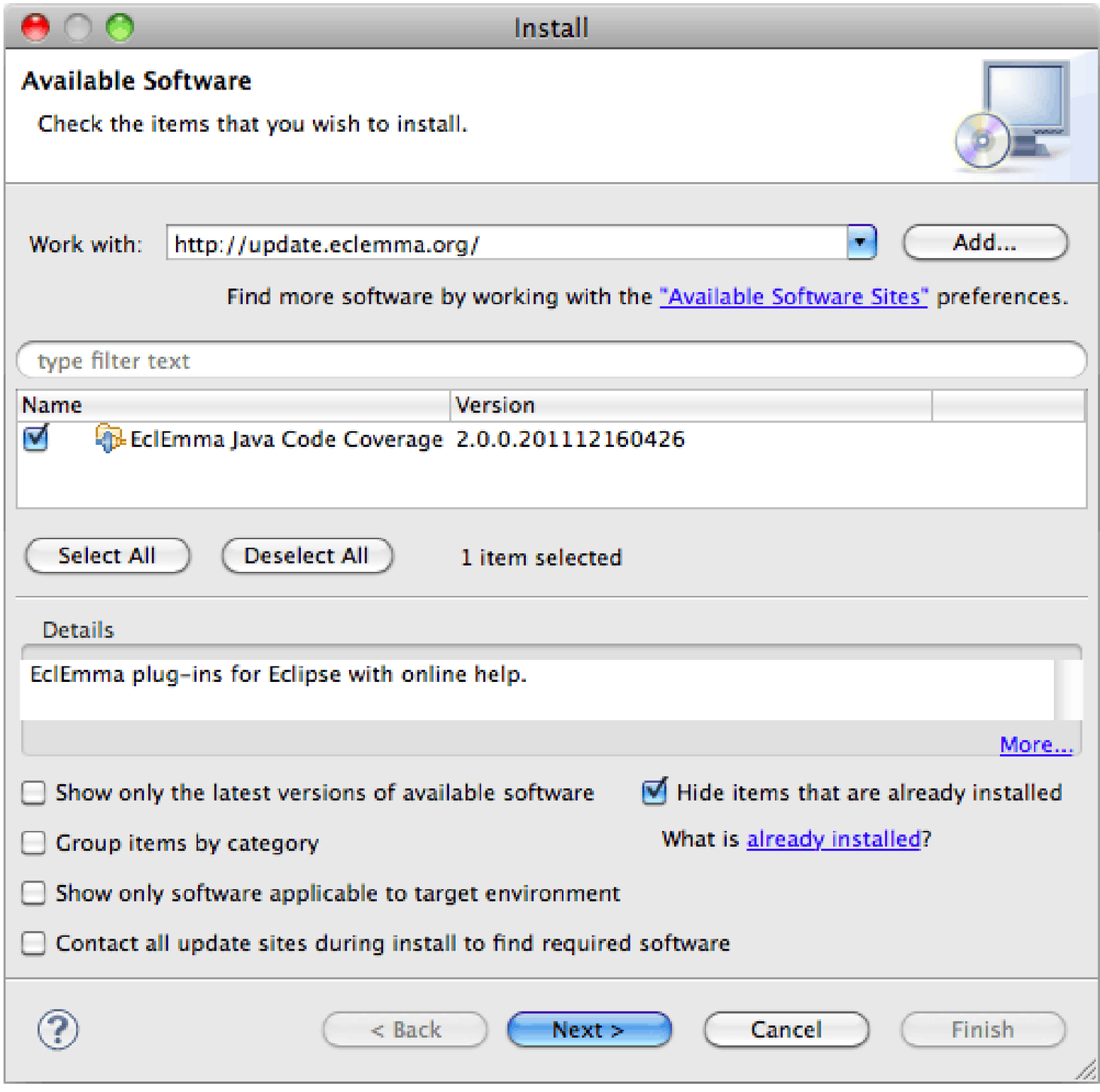
- Emma: http://emma.sourceforge.net/
- EclEmma: http://www.eclemma.org/installation.html/
- Cobertura: http://cobertura.github.io/cobertura/
- Clover: https://www.atlassian.com/software/clover/overview
- JCov: https://wiki.openjdk.java.net/display/CodeTools/jcov
- JaCoCo: http://www.eclemma.org/jacoco/

EclEmma: installation

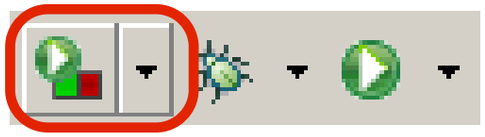
- From your Eclipse menu select Help → Install New Software...
- In the Install dialog enter http://update.eclemma.org/ at the Work with field
- Check the latest EclEmma version and press Next
- Follow the steps in the installation wizard.

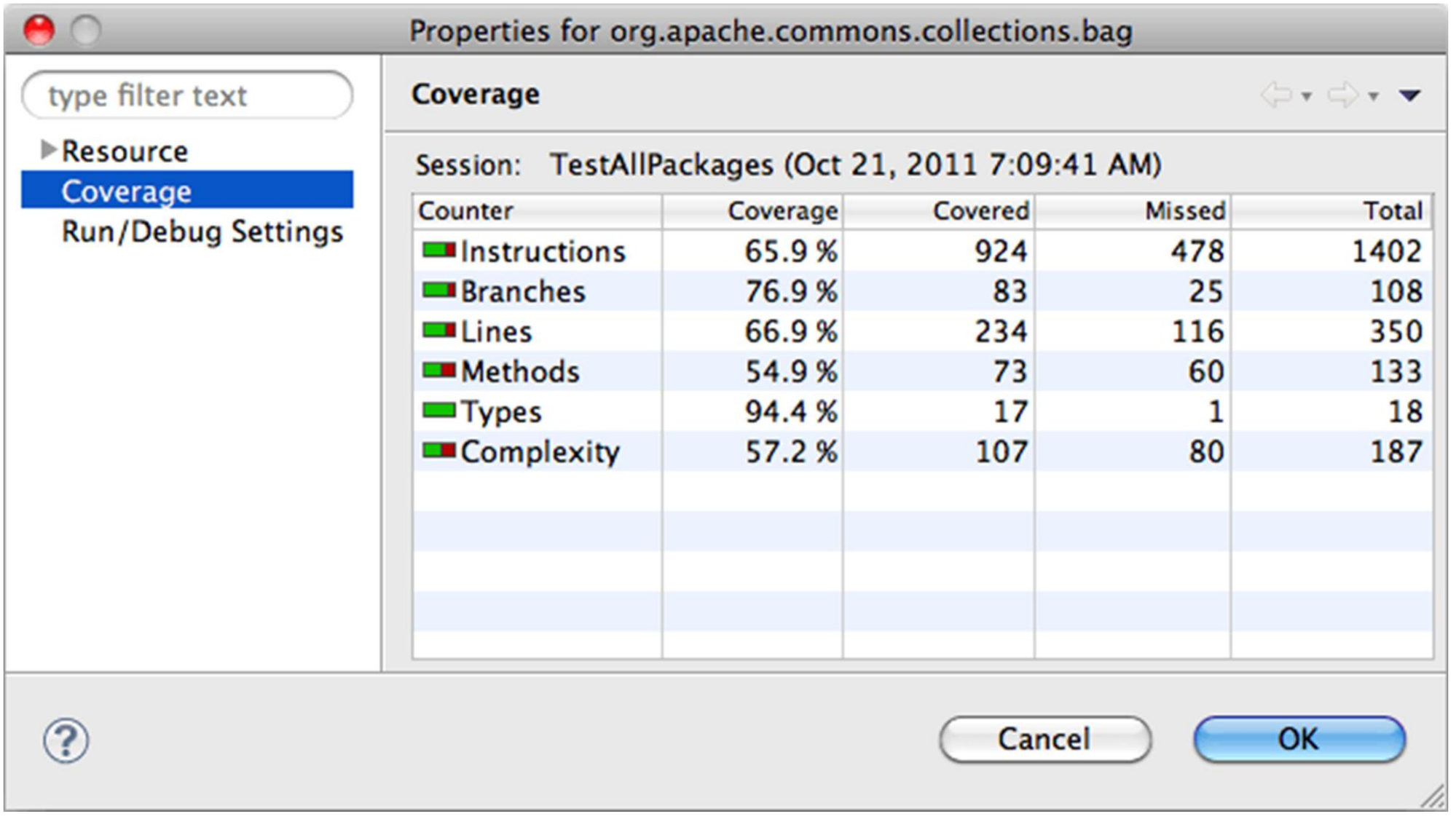
EclEmma: installation

- From your Eclipse menu select Help → Install New Software...
- In the Install dialog enter http://update.eclemma.org/ at the Work with field
- Check the latest EclEmma version and press Next
- Follow the steps in the installation wizard.

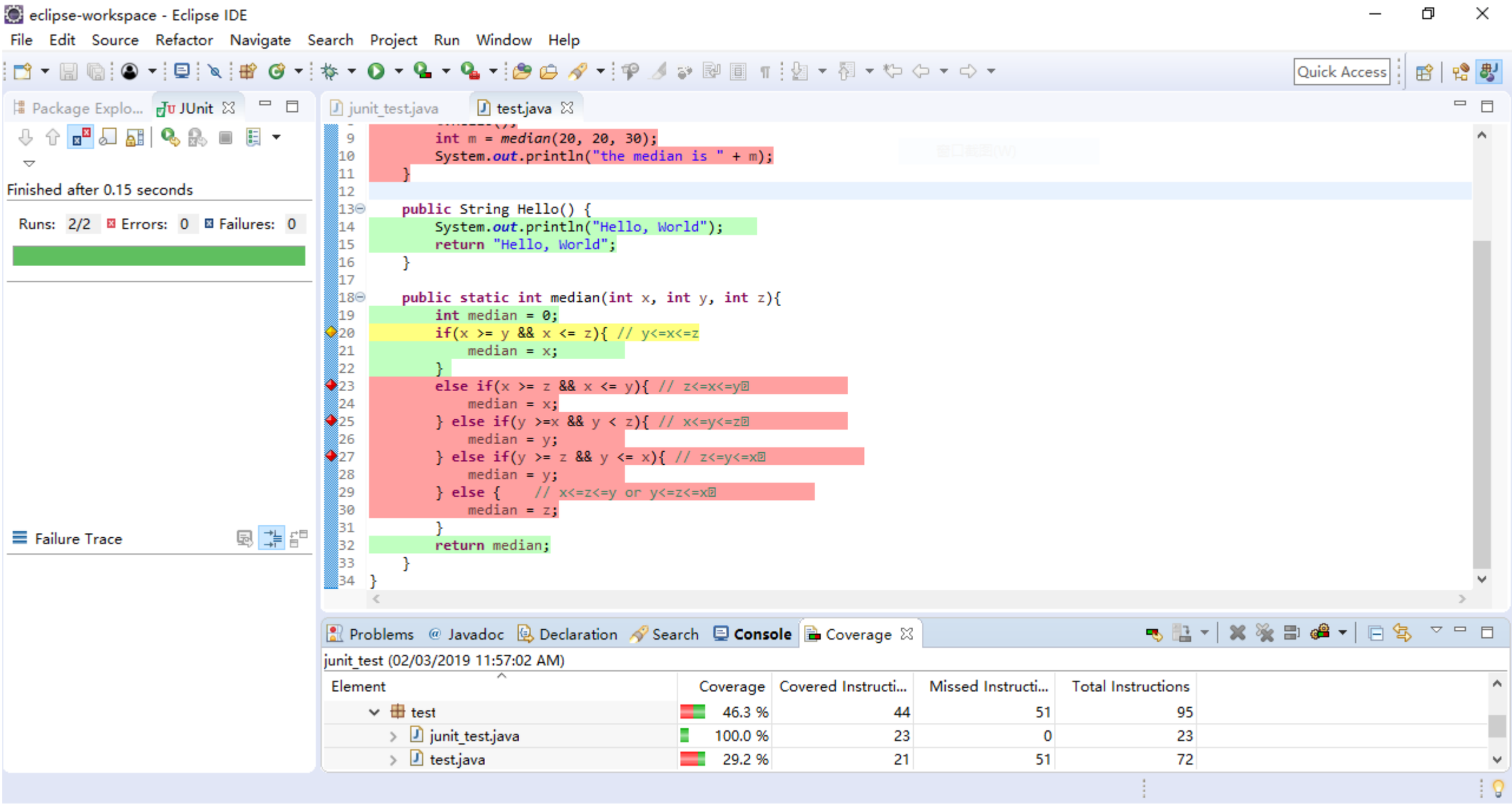


EclEmma: execution

- The installation was successful if you can see the coverage launcher in the toolbar of the Java perspective:
- Coverage collection 



EclEmma: Demo



JUnit – Online Resources

https://www.tutorialspoint.com/junit/junit_quick_guide.htm

http://www.vogella.com/tutorials/JUnit/article.html

https://help.eclipse.org/neon/index.jsp?topic=%2Forg.eclipse.jdt.doc.user%2FgettingStarted%2Fqs-junit.htm

www.junit.org

Prepare for testing—IntelliJ IDEA (jetbrains.com)

Thanks!


Testing
Software
design
articles
management
web
use
test
code
quality
development
Ruby
James
system
article
People
Project
information
check
Agile
testers
good
risk
usability
Tools
life
bug
time
approach
exploratory
introduction