Assert… methods in JUnit4

The JUnit4 testing system provides other methods besides assertEquals in order to test code. They are declared in a class Assert whose documentation is here:

http://junit.org/junit4/javadoc/latest/

We summarize the methods below, after a warning.

Do not use the assert statement to test in a JUnit class!

It is possible to use an assert statement like the following in a JUnit test class, and it will work. But do *not* do this. That statement is used for a different purpose. Instead, use only the methods shown in the table below.

```
assert n == 5;      // do not use this in a Junit testing class.
```

assert… methods in JUnit4

The table below gives a summary of the method calls you can use in a JUnit testing class. With each, to the right, we say when it "succeeds", meaning when it does not throw an exception. In all other cases, the call throws an AssertionError and the JUnit pane will show a red line, meaning there is an error.

When methods have two parameters, the first, written *exp*, is the expected value and the second, written *comp*, is the computed value. For example, if you want to test whether method p.getYear() returns 2017, use this:

```
assertEquals(2017, p.getYear());
```

Most methods are overloaded, allowing different parameter types. For some, we give three different versions, summarizing to the right what they mean. Note that special methods are used for floating-point types. It does not make sense to test whether two floating-point values are equal because a floating-point value is only an approximation to some real value.

For class types, assertEquals(exp, comp) uses function equals while assertSame(exp, comp) uses operator ==.

assertEquals(exp, comp)	Use for primitive types except floats. Succeed if exp equals comp.
assertEquals(exp, comp)	Use for class types. Succeed if exp.equals(comp)
assertEquals(exp, comp, delta)	Use for floats. Succeed if abs(exp – comp) < delta
assertNotEquals(exp, comp)	All methods as for assertEquals. Obvious meaning
assertArrayEquals(exp, comp)	All methods as for assertEquals. The parameters are arrays, and this succeeds if the arrays are equal
assertNotNull(comp)	Succeed if comp is not null
assertNull(comp)	Succeed if comp is null
assertNotSame(exp, comp)	exp and comp are class types. Succeed if exp != comp
assertSame(exp, comp)	exp and comp are class types. Succeed if exp == comp
assertFalse(comp)	Succeed if comp is false
assertTrue(comp)	Succeed if comp is true
fail()	Never succeed, always throw an AssertionError exception