OpenJML

**Command Line**

Installation

* Download OpenJML tar file from http://www.openjml.org/documentation/installation.shtml
* Extract file to folder /$ using command
  + tar xzvf openjml.tar.gz -C /$/openjml

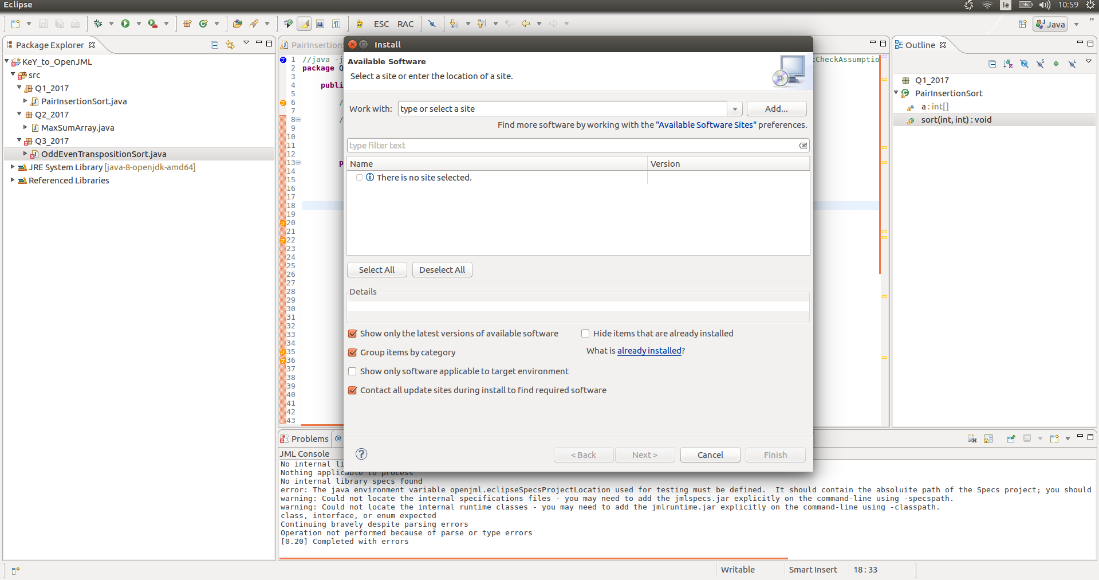
Commands for executing OpenJML from command line

* java –jar <$/openjml/openjml.jar (Required to run OpenJML alongside java file)
* -specspath <$/openjml/jmlspecs.jar> (Set the specification path for the OpenJML compiler)
* -option (Select the OpenJML task to want to perform, a list of such options is attached below)
* <$/file.java> (Select a java file)

**Eclipse**

Installed on Eclipse version 3.8.1 on Ubuntu 16.04

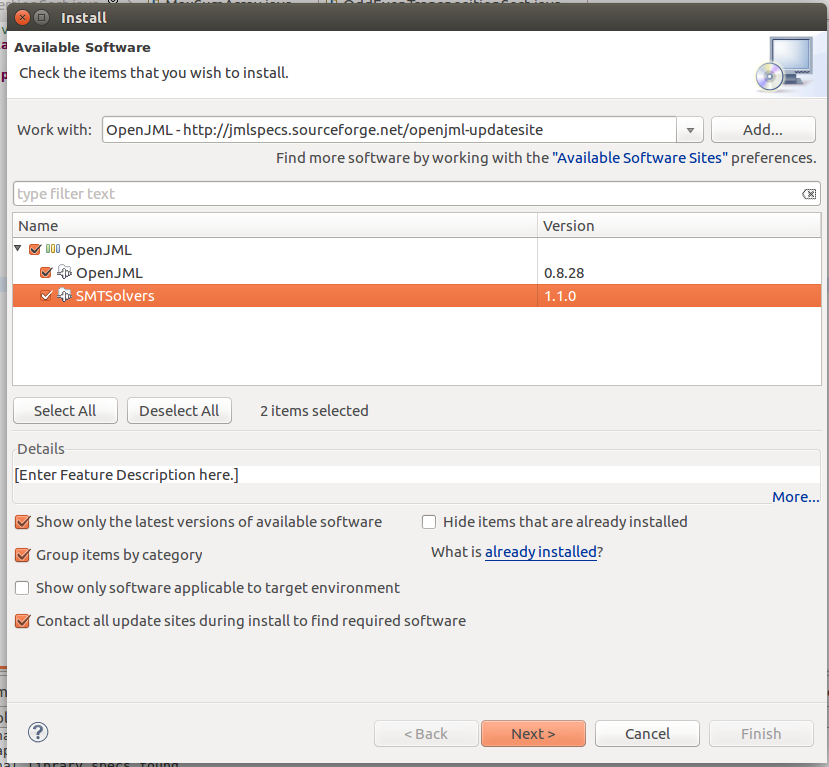
Go to Help – Install New Software



Insert <http://jmlspecs.sourceforge.net/openjml-updatesite>

Select the two options

* + OpenJML
  + SMTSolvers



Click Next

Agree to the disclaimer and select Finish to complete installation

A Coffee Cup, RAC and ESC icon will be added to the Eclipse banner



**OpenJML Testing**

Using two programs;

1. **PairInsertionSort.java**

package Q1\_2017;

public class PairInsertionSort {

/\*@ non\_null spec\_public @\*/ int[] a;

/\*@ public behaviour

@ requires 0 < l && l <= r && r < a.length;

@ assignable a[\*];

@ ensures (\forall int i; l <= i && i < r; a[i] <= a[i + 1]);

@\*/

public void sort(int l, int r) {

int left = l;

int right = r;

/\*@ assignable a[\*];

@ decreases right + 1 - left;

@ loop\_invariant l <= k && k <= right;

@ loop\_invariant l <= left && left <= right + 1 && right == r;

@ loop\_invariant (\forall int i; l <= i && i < left; a[i] <= a[i + 1]);

@\*/

for (int k = left; ++left <= right; k = ++left)

{

int a1 = a[k];

int a2 = a[left];

if (a1 < a2) {

a2 = a1; a1 = a[left];

}

/\*@

@ assignable a[\*];

@ decreases k;

@ loop\_invariant l <= k && k < r;

@ loop\_invariant (\forall int i; l <= i && i < k-1; a[i] <= a[i + 1]);

@\*/

while (a1 < a[--k]) {

a[k + 2] = a[k];

}

a[++k + 1] = a1;

/\*@

@ assignable a[\*];

@ decreases k;

@ loop\_invariant l <= k && k < r;

@ loop\_invariant (\forall int i; l <= i && i < k-1; a[i] <= a[i + 1]);

@\*/

while (a2 < a[--k]) {

a[k + 1] = a[k];

}

a[k + 1] = a2;

}

int last = a[right];

/\*@

@ assignable a[\*];

@ decreases right;

@ loop\_invariant l <= right && right < r;

@ loop\_invariant right <= left + 1;

@ loop\_invariant (\forall int i; right <= i && i <= r; last <= a[i]);

@ loop\_invariant (\forall int i; l <= i && i < right - 1; a[i] <= a[i + 1]);

@ loop\_invariant (\forall int i; right < i && i < r-1; a[i] <= a[i + 1]);

@\*/

while (last < a[--right]) {

a[right + 1] = a[right];

}

a[right + 1] = last;

}

}

1. **MaxSumArray.java**

package Q2\_2017;

public class MaxSumArray {

/\*

\* Managed to show permutation property.

\* Started working on the sortedness property, I think I was on the right track.

\* Didn't work on termination at all.

\*

\* \*/

/\*@ public normal\_behaviour

requires 0 <= i && i < a.length;

requires 0 <= j && j < a.length;

ensures \old(a[i]) == a[j];

ensures \old(a[j]) == a[i];

assignable a[i], a[j];

@\*/

//ensures \dl\_seqPerm(\dl\_array2seq(a), \old(\dl\_array2seq(a)));

public void swap(int[] a, int i, int j){

int temp = a[i];

a[i] = a[j];

a[j] = temp;

}

/\*@ public normal\_behaviour

ensures (\forall int j; 0 <= j && j < a.length-1; a[j] <= a[j+1]);

diverges true;

@\*/

//ensures \dl\_seqPerm(\dl\_array2seq(a), \old(\dl\_array2seq(a)));

public void sort(int[] a){

boolean sorted = false;

/\*

@ loop\_invariant sorted ==> (\forall int j; 0 <= j && j < a.length-1; a[j] <= a[j+1]);

@\*/

// loop\_invariant \dl\_seqPerm(\dl\_array2seq(a), \old(\dl\_array2seq(a)));

while(!sorted){

sorted = true;

/\*@

@ assignable a[\*];

@ loop\_invariant 1 <= j && (j <= a.length || a.length == 0);

@ loop\_invariant sorted ==> (\forall int i; 0 <= i && 1 + 2 \* i < j; a[2\*i+1] <= a[2\*i+2]);

@\*/

// loop\_invariant \dl\_seqPerm(\dl\_array2seq(a), \old(\dl\_array2seq(a)));

for(int j = 1; j < a.length-1; j+=2){

if(a[j] > a[j+1]){

swap(a, j, j+1);

sorted = false;

}

}

/\* @ assignable a[\*];

@ loop\_invariant 0 <= k && k <= a.length;

@ loop\_invariant sorted ==> (\forall int i; 0 <= i && 2 \* i < k; a[2\*i] <= a[2\*i+1]) && (\forall int i; 0 <= i && 1+ 2 \* i < a.length-1; a[2\*i+1] <= a[2\*i+2]);

@ loop\_invariant \dl\_seqPerm(\dl\_array2seq(a), \old(\dl\_array2seq(a)));

@\*/

for(int k = 0; k < a.length-1; k+=2){

if(a[k] > a[k+1]){

swap(a, k, k+1);

sorted = false;

}

}

}

}

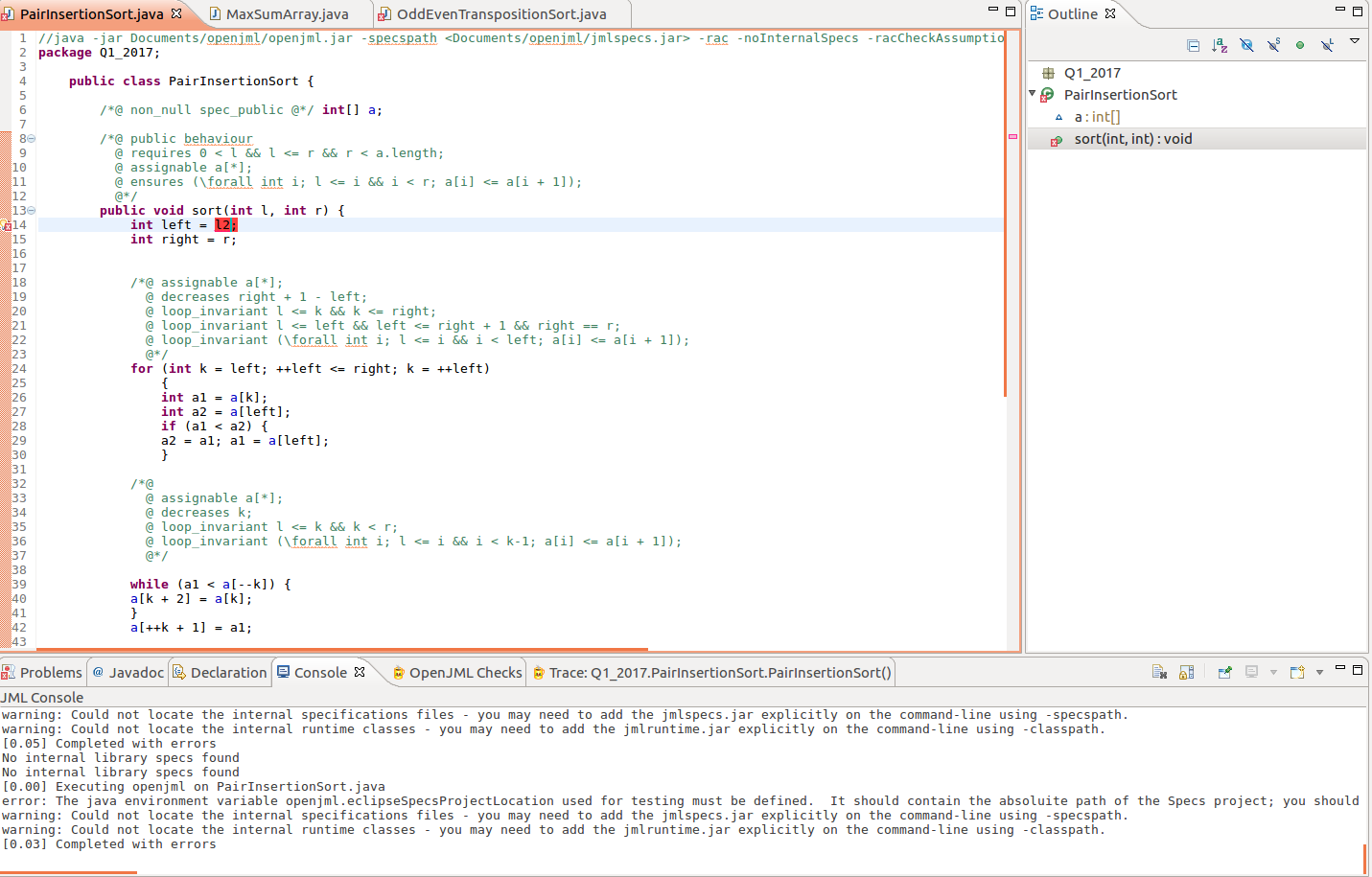
**Type checking**

1. Eclipse – Coffee Cup

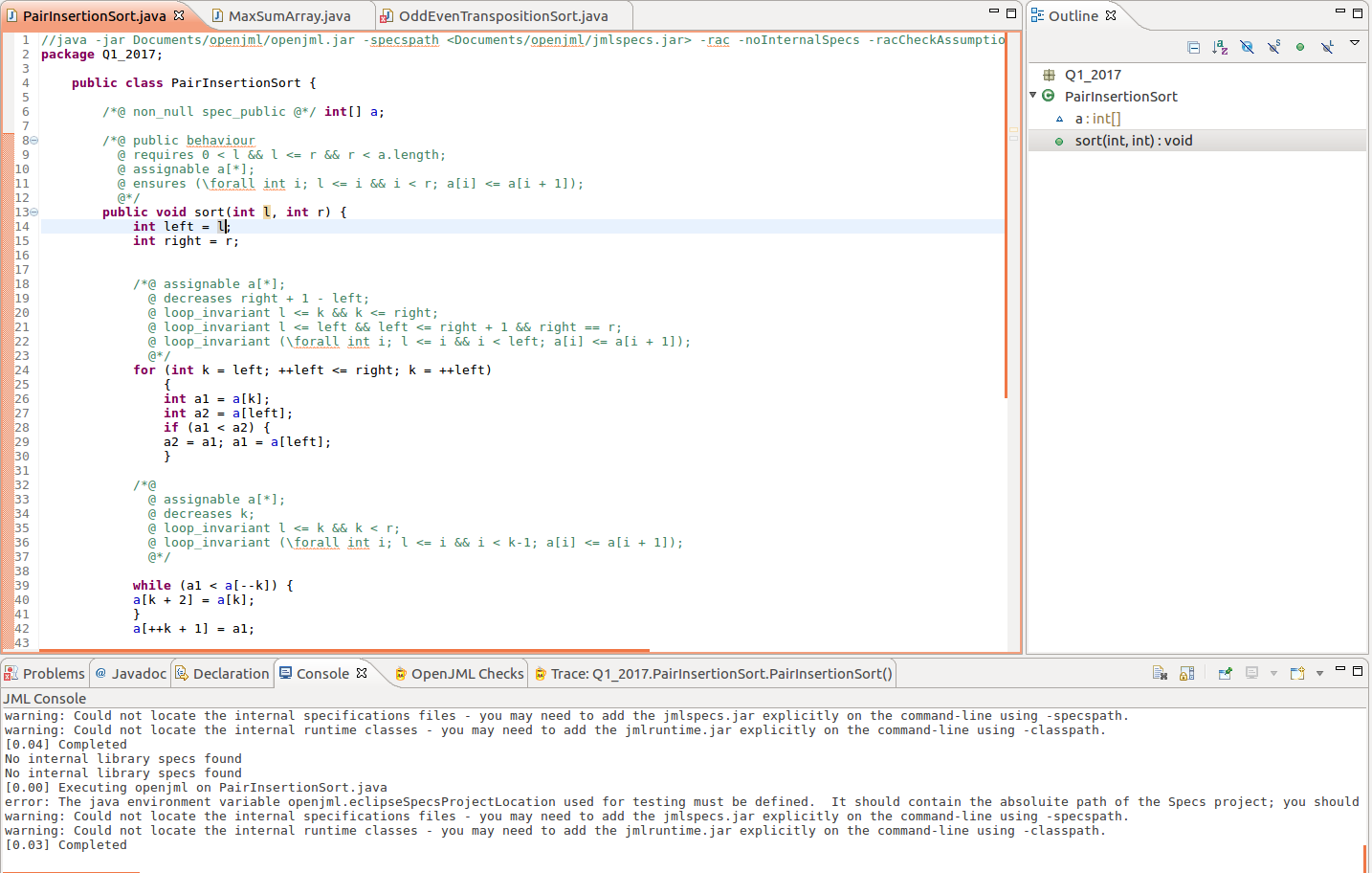
* Open any Java program and select the Coffee-Cup icon

When first run on a program with type-check errors, the issues are highlighted in red.

In the example below, the assignment of the variable 'left' to 'l' in the method 'sort' which has been changed to assign to variable 'l2' resulting in an error due to no known variable 'l2' being found.



When the variable 'l2' is renamed back to 'l', the type-checking passes, and no errors are reported



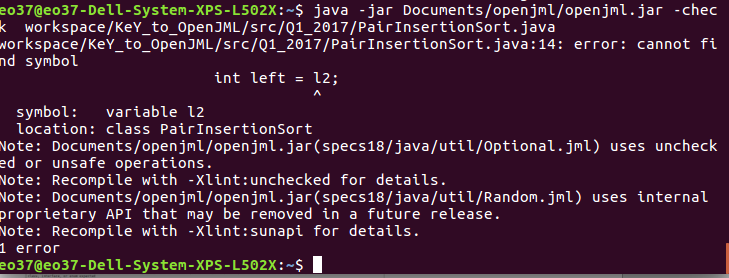
1. Command Line

* Run the –check option to perform type checking from the command line

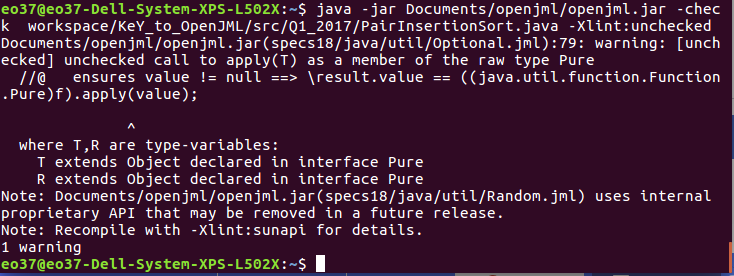
We again change the variable 'l' in the method 'sort' to 'l2' during the assignment to the variable 'left'.

The line number (14) and error itself are displayed with the symbol and class name also shown.

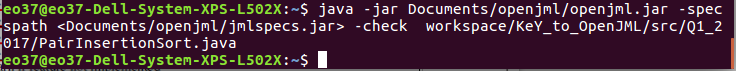
However, the method name where the error occurred is not shown, which would be useful.



Running the Xlint:unchecked option results in polymorphism failing, which is a Java issue and not an OpenJML issue.

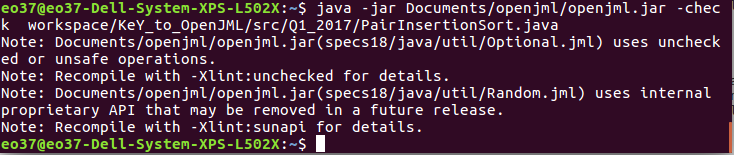


NO ERROR WAS FOUND FROM COMMAND LINE WHEN THE SPECSPATH IS SET?



Again, once the variable is renamed back to 'l', no issues are found.

Note: No message appears to confirm that the type-checking has been performed and passed successfully.



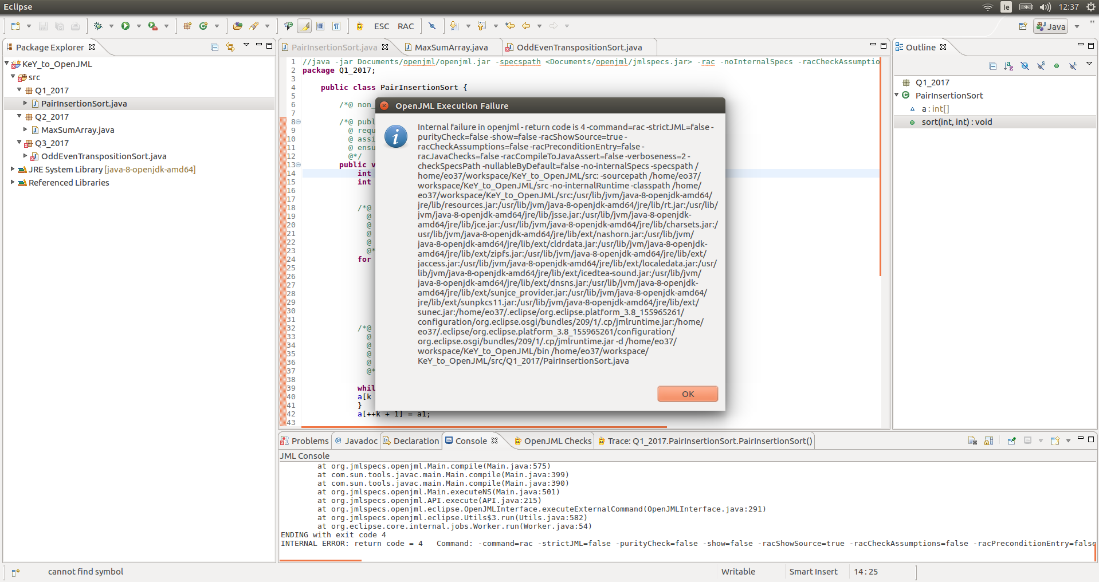
**Runtime Assertion Checking (RAC)**

Eclipse

* + Once the type-checking is completed, Runtime Assertion Checking (RAC) can be performed
  + Select the RAC icon

1. PairInsertionSort

ERROR: RAC Internal Error occurred on first run against program PairInsertionSort.java



[0.00] Executing openjml on PairInsertionSort.java

error: The java environment variable openjml.eclipseSpecsProjectLocation used for testing must be defined. It should contain the absoluite path of the Specs project; you should set it as a VM argument in run Run Configuration for tests, as in -Dopenjml.eclipseSpecsProjectLocation=...

warning: Could not locate the internal specifications files - you may need to add the jmlspecs.jar explicitly on the command-line using -specspath.

warning: Could not locate the internal runtime classes - you may need to add the jmlruntime.jar explicitly on the command-line using -classpath.

RAC-Compiling Q1\_2017.PairInsertionSort

Internal JML bug - please report. BuildOpenJML-20180326

java.lang.AssertionError: \_JML\_\_tmp21

at com.sun.tools.javac.comp.Lower.unbox(Lower.java:3177)

at com.sun.tools.javac.comp.Lower.boxIfNeeded(Lower.java:3136)

at com.sun.tools.javac.comp.Lower.translate(Lower.java:2382)

at com.sun.tools.javac.comp.Lower.visitBinary(Lower.java:3367)

at com.sun.tools.javac.tree.JCTree$JCBinary.accept(JCTree.java:1785)

at com.sun.tools.javac.tree.TreeTranslator.translate(TreeTranslator.java:58)

at com.sun.tools.javac.comp.Lower.translate(Lower.java:2371)

at com.sun.tools.javac.comp.Lower.translate(Lower.java:2382)

at com.sun.tools.javac.comp.Lower.visitVarDef(Lower.java:3547)

at com.sun.tools.javac.tree.JCTree$JCVariableDecl.accept(JCTree.java:852)

at org.jmlspecs.openjml.JmlTree$JmlVariableDecl.accept(JmlTree.java:1284)

at com.sun.tools.javac.tree.TreeTranslator.translate(TreeTranslator.java:58)

at com.sun.tools.javac.comp.Lower.translate(Lower.java:2371)

at com.sun.tools.javac.tree.TreeTranslator.translate(TreeTranslator.java:70)

at com.sun.tools.javac.tree.TreeTranslator.visitBlock(TreeTranslator.java:162)

at com.sun.tools.javac.comp.Lower.visitBlock(Lower.java:3561)

at com.sun.tools.javac.tree.JCTree$JCBlock.accept(JCTree.java:909)

at org.jmlspecs.openjml.JmlTree$JmlBlock.accept(JmlTree.java:1227)

at com.sun.tools.javac.tree.TreeTranslator.translate(TreeTranslator.java:58)

at com.sun.tools.javac.comp.Lower.translate(Lower.java:2371)

at com.sun.tools.javac.comp.Lower.visitWhileLoop(Lower.java:3573)

at com.sun.tools.javac.tree.JCTree$JCWhileLoop.accept(JCTree.java:965)

at org.jmlspecs.openjml.JmlTree$JmlWhileLoop.accept(JmlTree.java:1716)

at com.sun.tools.javac.tree.TreeTranslator.translate(TreeTranslator.java:58)

at com.sun.tools.javac.comp.Lower.translate(Lower.java:2371)

at com.sun.tools.javac.tree.TreeTranslator.translate(TreeTranslator.java:70)

at com.sun.tools.javac.tree.TreeTranslator.visitBlock(TreeTranslator.java:162)

at com.sun.tools.javac.comp.Lower.visitBlock(Lower.java:3561)

at com.sun.tools.javac.tree.JCTree$JCBlock.accept(JCTree.java:909)

at org.jmlspecs.openjml.JmlTree$JmlBlock.accept(JmlTree.java:1227)

at com.sun.tools.javac.tree.TreeTranslator.translate(TreeTranslator.java:58)

at com.sun.tools.javac.comp.Lower.translate(Lower.java:2371)

at com.sun.tools.javac.tree.TreeTranslator.translate(TreeTranslator.java:70)

at com.sun.tools.javac.tree.TreeTranslator.visitBlock(TreeTranslator.java:162)

at com.sun.tools.javac.comp.Lower.visitBlock(Lower.java:3561)

at com.sun.tools.javac.tree.JCTree$JCBlock.accept(JCTree.java:909)

at org.jmlspecs.openjml.JmlTree$JmlBlock.accept(JmlTree.java:1227)

at com.sun.tools.javac.tree.TreeTranslator.translate(TreeTranslator.java:58)

at com.sun.tools.javac.comp.Lower.translate(Lower.java:2371)

at com.sun.tools.javac.tree.TreeTranslator.visitTry(TreeTranslator.java:218)

at com.sun.tools.javac.comp.Lower.visitTry(Lower.java:3897)

at com.sun.tools.javac.tree.JCTree$JCTry.accept(JCTree.java:1173)

at com.sun.tools.javac.tree.TreeTranslator.translate(TreeTranslator.java:58)

at com.sun.tools.javac.comp.Lower.translate(Lower.java:2371)

at com.sun.tools.javac.tree.TreeTranslator.translate(TreeTranslator.java:70)

at com.sun.tools.javac.tree.TreeTranslator.visitBlock(TreeTranslator.java:162)

at com.sun.tools.javac.comp.Lower.visitBlock(Lower.java:3561)

at com.sun.tools.javac.tree.JCTree$JCBlock.accept(JCTree.java:909)

at org.jmlspecs.openjml.JmlTree$JmlBlock.accept(JmlTree.java:1227)

at com.sun.tools.javac.tree.TreeTranslator.translate(TreeTranslator.java:58)

at com.sun.tools.javac.comp.Lower.translate(Lower.java:2371)

at com.sun.tools.javac.tree.TreeTranslator.visitMethodDef(TreeTranslator.java:145)

at com.sun.tools.javac.comp.Lower.visitMethodDefInternal(Lower.java:2828)

at com.sun.tools.javac.comp.Lower.visitMethodDef(Lower.java:2737)

at com.sun.tools.javac.tree.JCTree$JCMethodDecl.accept(JCTree.java:778)

at org.jmlspecs.openjml.JmlTree$JmlMethodDecl.accept(JmlTree.java:1158)

at com.sun.tools.javac.tree.TreeTranslator.translate(TreeTranslator.java:58)

at com.sun.tools.javac.comp.Lower.translate(Lower.java:2371)

at com.sun.tools.javac.comp.Lower.visitClassDef(Lower.java:2508)

at com.sun.tools.javac.tree.JCTree$JCClassDecl.accept(JCTree.java:693)

at org.jmlspecs.openjml.JmlTree$JmlClassDecl.accept(JmlTree.java:1074)

at com.sun.tools.javac.tree.TreeTranslator.translate(TreeTranslator.java:58)

at com.sun.tools.javac.comp.Lower.translate(Lower.java:2371)

at com.sun.tools.javac.comp.Lower.translate(Lower.java:2390)

at com.sun.tools.javac.comp.Lower.translateTopLevelClass(Lower.java:3932)

at com.sun.tools.javac.main.JavaCompiler.desugar(JavaCompiler.java:1522)

at org.jmlspecs.openjml.JmlCompiler.desugar(JmlCompiler.java:434)

at com.sun.tools.javac.main.JavaCompiler.compile2(JavaCompiler.java:898)

at org.jmlspecs.openjml.JmlCompiler.compile2(JmlCompiler.java:674)

at com.sun.tools.javac.main.JavaCompiler.compile(JavaCompiler.java:867)

at com.sun.tools.javac.main.Main.compile(Main.java:553)

at com.sun.tools.javac.main.Main.compile(Main.java:410)

at org.jmlspecs.openjml.Main.compile(Main.java:575)

at com.sun.tools.javac.main.Main.compile(Main.java:399)

at com.sun.tools.javac.main.Main.compile(Main.java:390)

at org.jmlspecs.openjml.Main.executeNS(Main.java:501)

at org.jmlspecs.openjml.API.execute(API.java:215)

at org.jmlspecs.openjml.eclipse.OpenJMLInterface.executeExternalCommand(OpenJMLInterface.java:291)

at org.jmlspecs.openjml.eclipse.Utils$3.run(Utils.java:582)

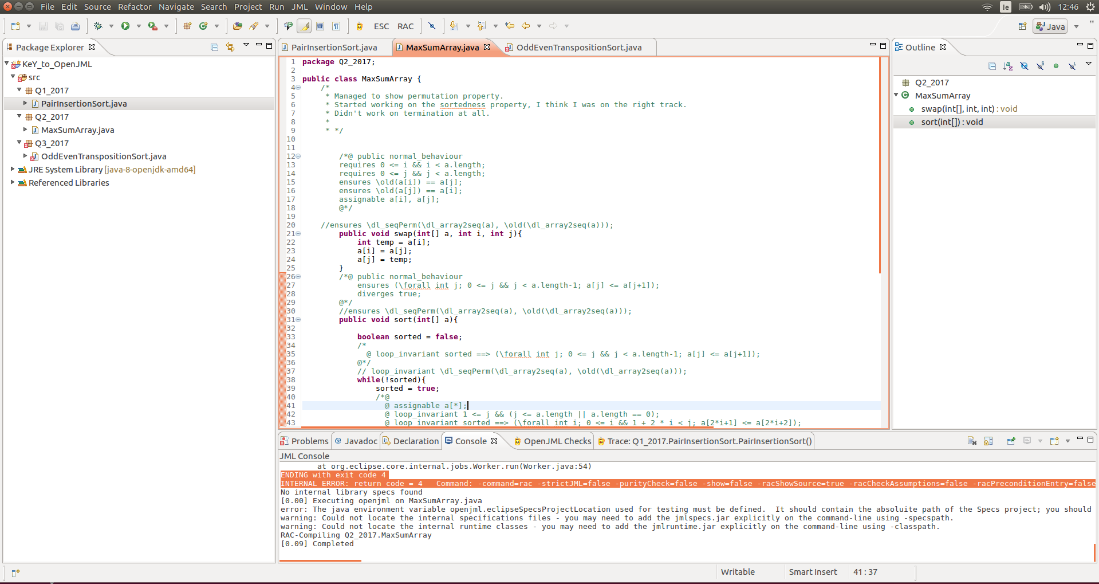
at org.eclipse.core.internal.jobs.Worker.run(Worker.java:54)

ENDING with exit code 4

ENDING with exit code 4

INTERNAL ERROR: return code = 4 Command: -command=rac -strictJML=false -purityCheck=false -show=false -racShowSource=true -racCheckAssumptions=false -racPreconditionEntry=false -racJavaChecks=false -racCompileToJavaAssert=false -verboseness=2 -checkSpecsPath -nullableByDefault=false -no-internalSpecs -specspath /home/eo37/workspace/KeY\_to\_OpenJML/src: -sourcepath /home/eo37/workspace/KeY\_to\_OpenJML/src -no-internalRuntime -classpath /home/eo37/workspace/KeY\_to\_OpenJML/src:/usr/lib/jvm/java-8-openjdk-amd64/jre/lib/resources.jar:/usr/lib/jvm/java-8-openjdk-amd64/jre/lib/rt.jar:/usr/lib/jvm/java-8-openjdk-amd64/jre/lib/jsse.jar:/usr/lib/jvm/java-8-openjdk-amd64/jre/lib/jce.jar:/usr/lib/jvm/java-8-openjdk-amd64/jre/lib/charsets.jar:/usr/lib/jvm/java-8-openjdk-amd64/jre/lib/ext/nashorn.jar:/usr/lib/jvm/java-8-openjdk-amd64/jre/lib/ext/cldrdata.jar:/usr/lib/jvm/java-8-openjdk-amd64/jre/lib/ext/zipfs.jar:/usr/lib/jvm/java-8-openjdk-amd64/jre/lib/ext/jaccess.jar:/usr/lib/jvm/java-8-openjdk-amd64/jre/lib/ext/localedata.jar:/usr/lib/jvm/java-8-openjdk-amd64/jre/lib/ext/icedtea-sound.jar:/usr/lib/jvm/java-8-openjdk-amd64/jre/lib/ext/dnsns.jar:/usr/lib/jvm/java-8-openjdk-amd64/jre/lib/ext/sunjce\_provider.jar:/usr/lib/jvm/java-8-openjdk-amd64/jre/lib/ext/sunpkcs11.jar:/usr/lib/jvm/java-8-openjdk-amd64/jre/lib/ext/sunec.jar:/home/eo37/.eclipse/org.eclipse.platform\_3.8\_155965261/configuration/org.eclipse.osgi/bundles/209/1/.cp/jmlruntime.jar:/home/eo37/.eclipse/org.eclipse.platform\_3.8\_155965261/configuration/org.eclipse.osgi/bundles/209/1/.cp/jmlruntime.jar -d /home/eo37/workspace/KeY\_to\_OpenJML/bin /home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java

1. MaxSumArray
   * No error when RAC is run against MaxSumArray.java (Unknown reason for error with first program)



[0.00] Executing openjml on MaxSumArray.java

error: The java environment variable openjml.eclipseSpecsProjectLocation used for testing must be defined. It should contain the absoluite path of the Specs project; you should set it as a VM argument in run Run Configuration for tests, as in -Dopenjml.eclipseSpecsProjectLocation=...

warning: Could not locate the internal specifications files - you may need to add the jmlspecs.jar explicitly on the command-line using -specspath.

warning: Could not locate the internal runtime classes - you may need to add the jmlruntime.jar explicitly on the command-line using -classpath.

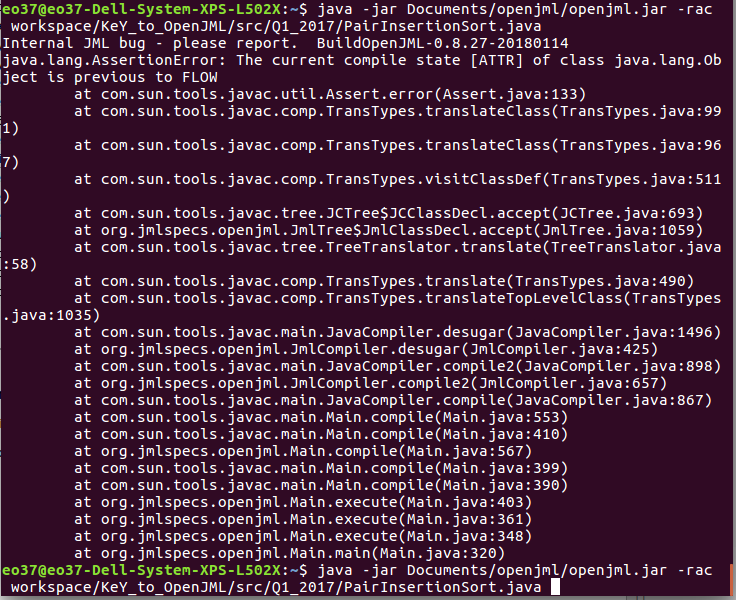
RAC-Compiling Q2\_2017.MaxSumArray

[0.09] Completed

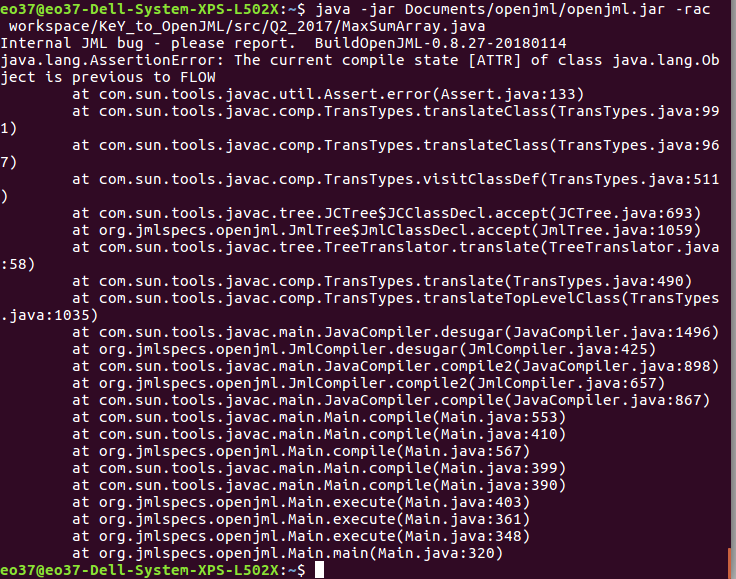
Command Line

* -rac option

1. PairInsertionSort
   * ERROR: RAC Error when run on program PairInsertionSort.java

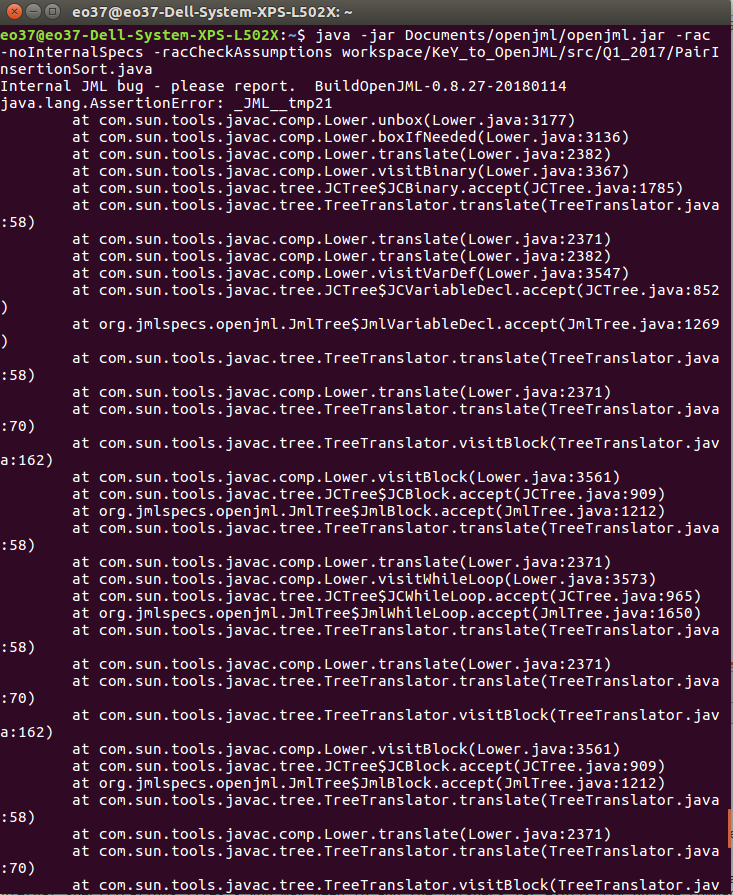


1. MaxSumArray
   * ERROR: RAC Error when run on program MaxSumArray.java

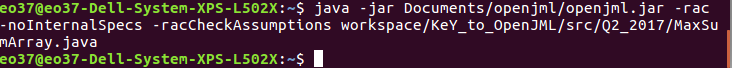


* -rac -noInternalSpecs -racCheckAssumptions options

1. PairInsertionSort
   * ERROR: RAC Error when run on program PairInsertionSort.java

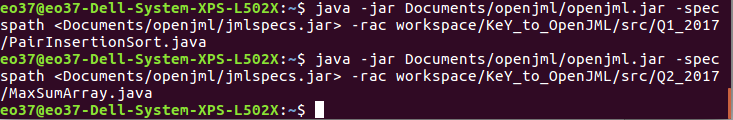


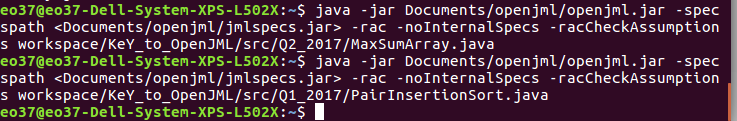
1. MaxSumArray
   * No error when run on program MaxSumArray.java
   * However, again no message to say RAC completed and was successful



NOTE:

* Inserting the specspath again removes all errors but unknown as to why and what is happening





**Extended Static Checking (ESC)**

Eclipse

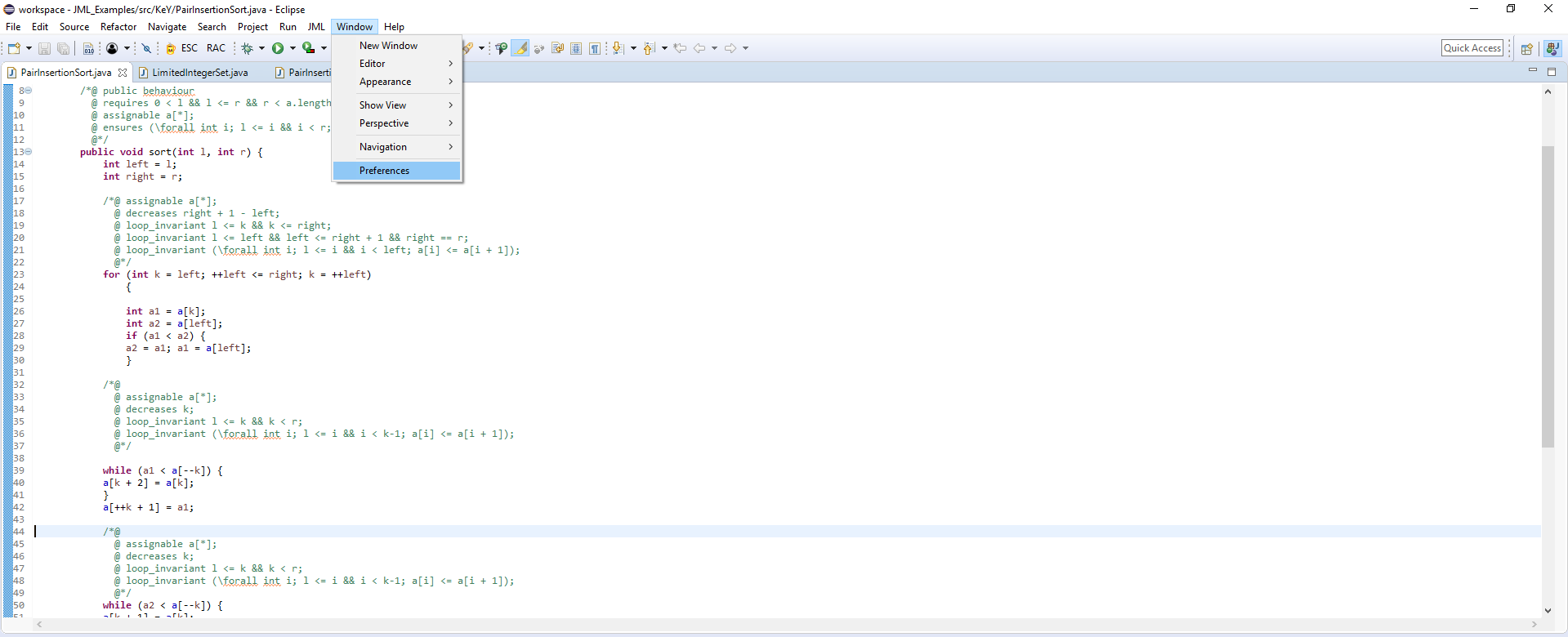
* Once the Runtime Assertion Checking is finished, ESC can be performed
* Select the ESC icon

1. PairInsertionSort

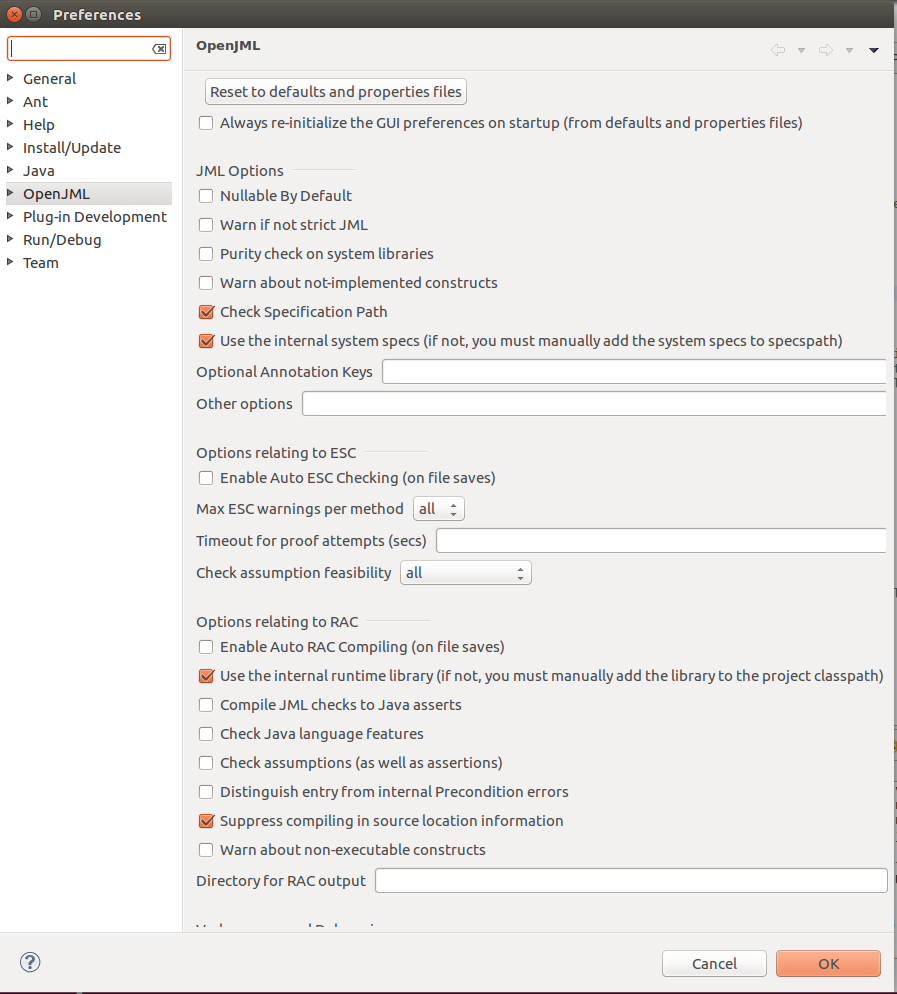
* **ERROR:**
  + Selecting the ESC button does not appear to be doing anything.
  + Have to set the OpenJML preferences to run ESC on every save

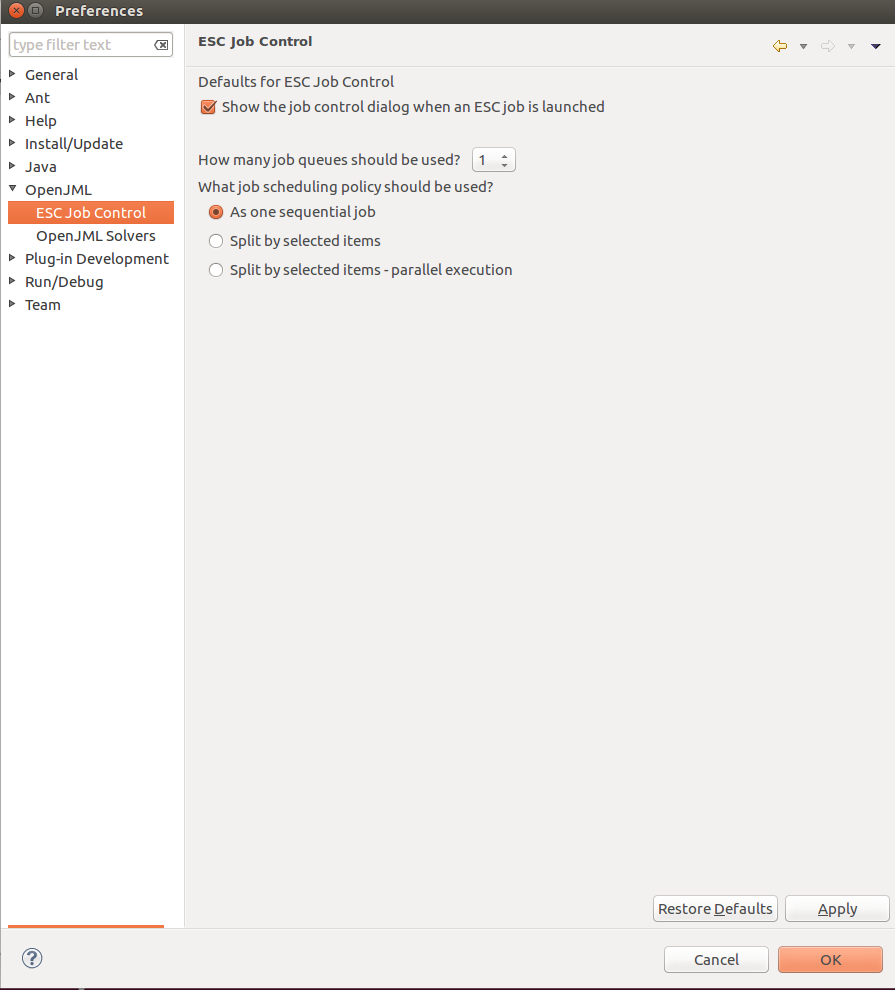
OpenJML Preferences (Eclipse)

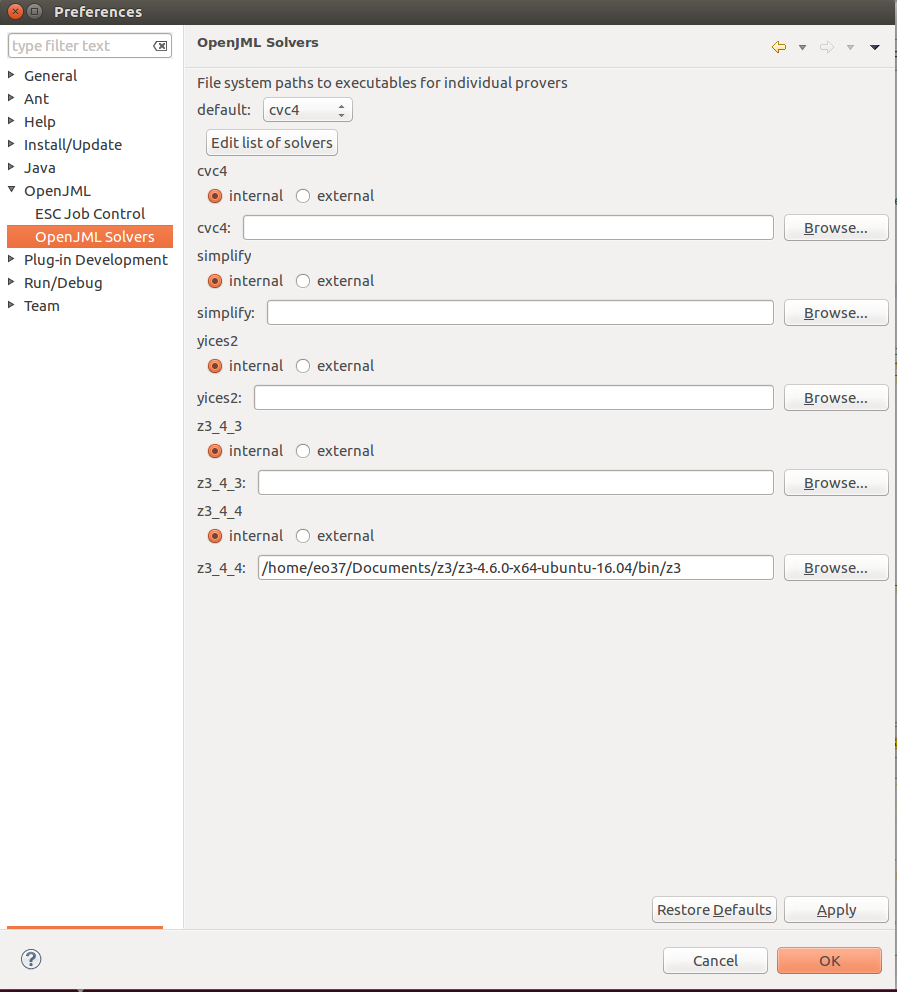
* Click on Window in menu bar
* Scroll down and select Preferences



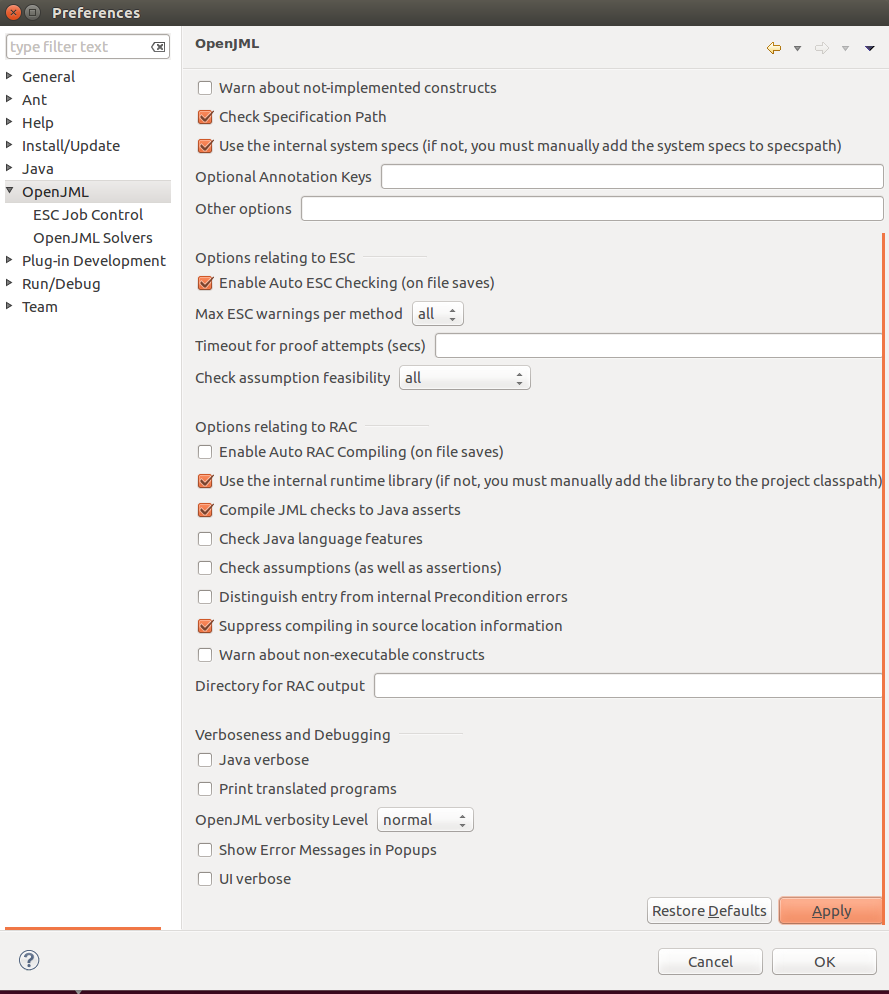
* Open OpenJML section with default settings
* Changes applied here will determine how the OpenJML system works.

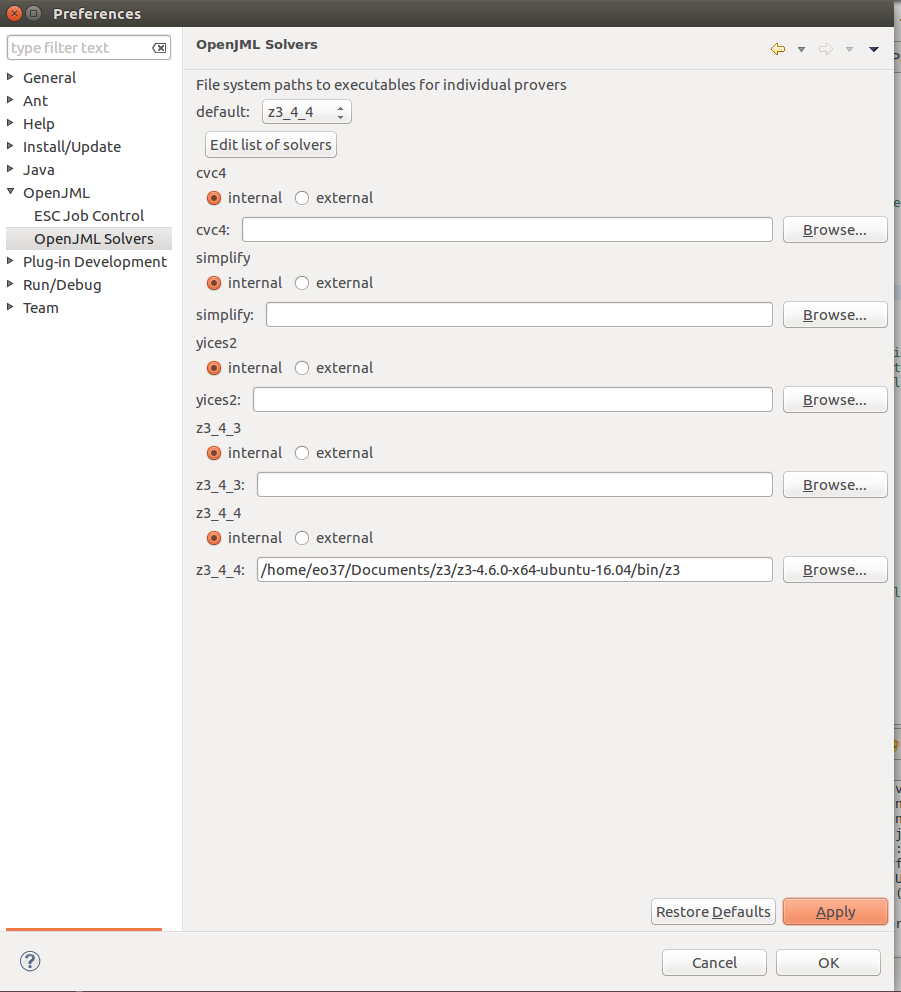




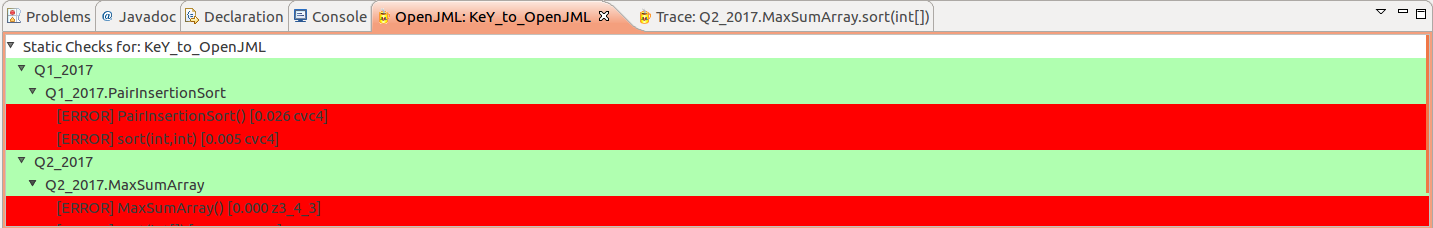


* We changed the default setting to enable ESC auto-checking once a save was made to a file.
* We also check assumptions as well as assertions and complied JML Checks to Java Assertions for one of the tests.
* We change the default prover to z3\_4\_4 to match the prover we have setup

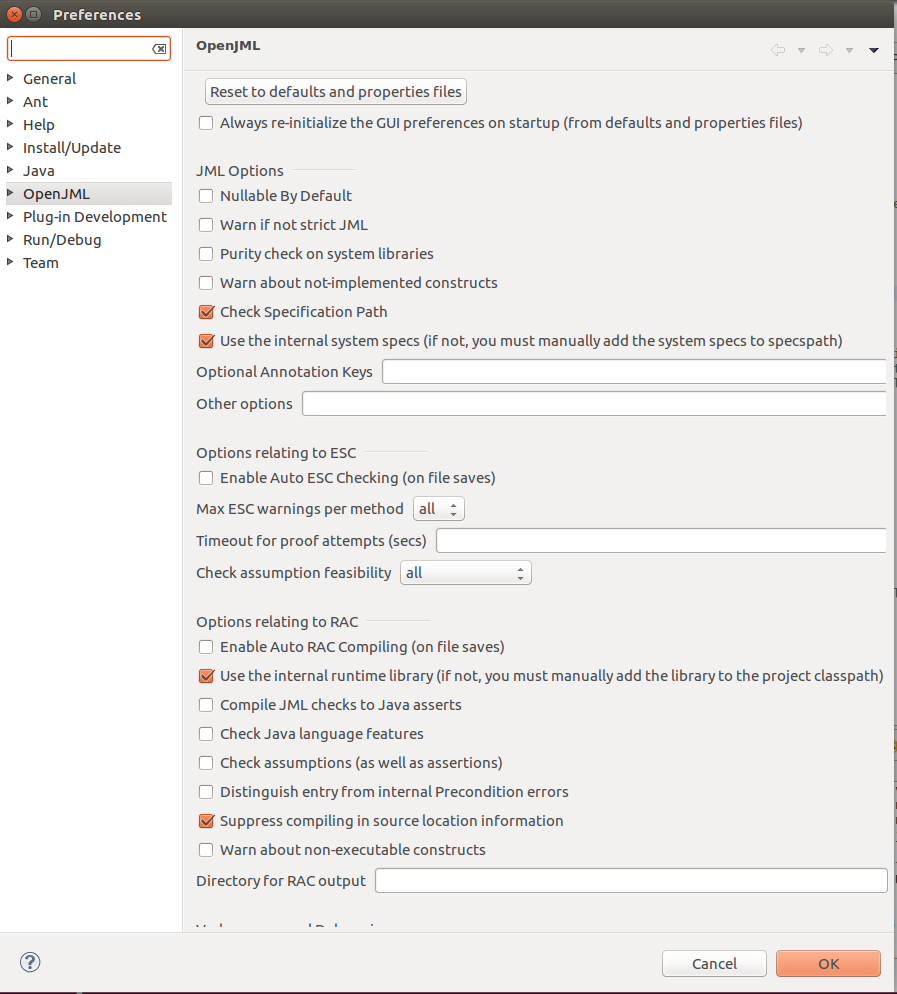


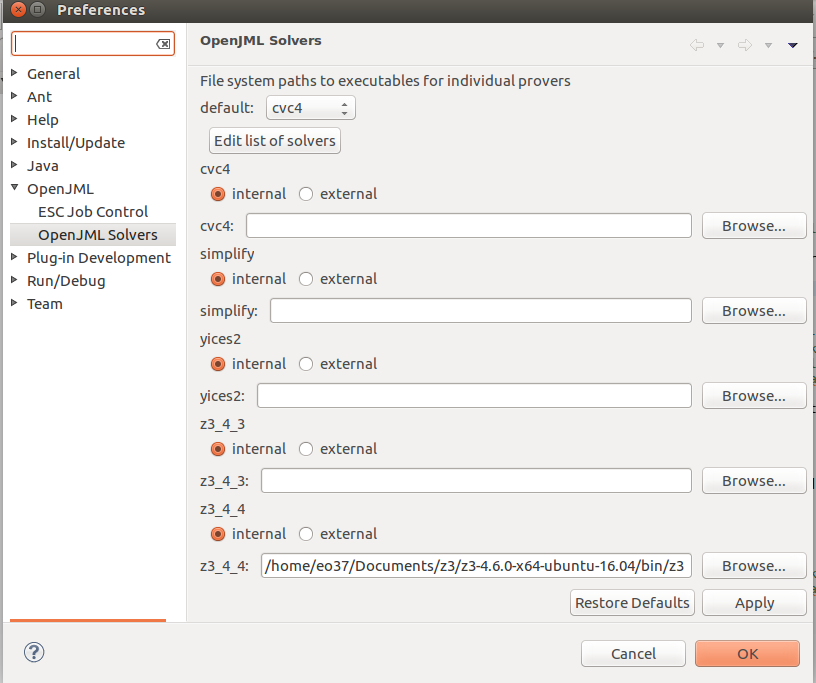


* Despite specifically specifying z3\_4\_4 as the prover for both programs, the static checker did not use it resulting in an error as the prover it tried to use could not be found.

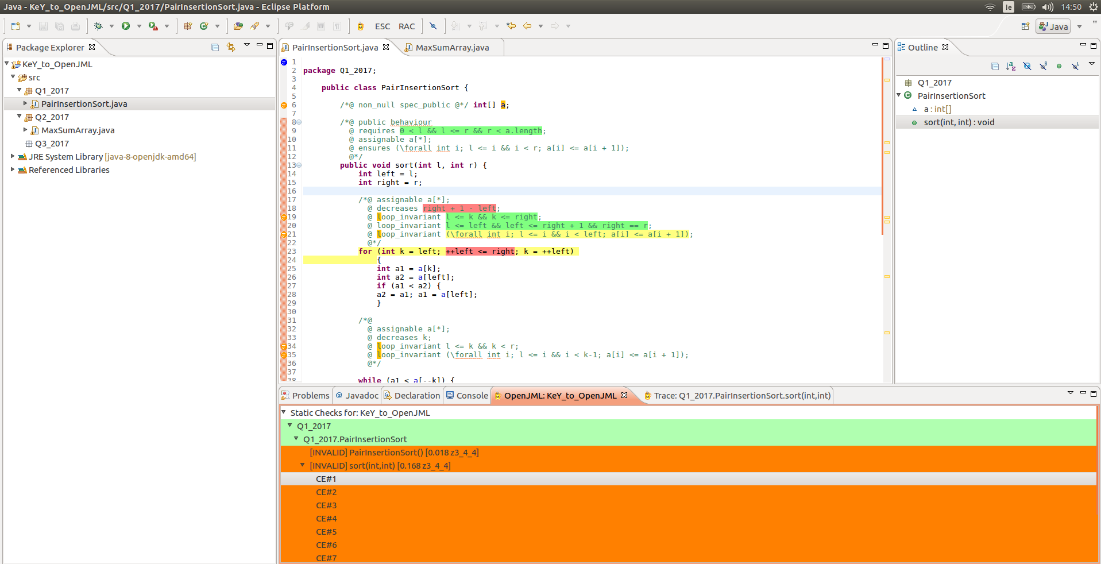


* Also after a couple of minutes the settings in the OpenJML preferences are reset to default and all the changes made are removed automatically which is frustrating as they must be setup again.





* The solution to fixing the prover issue is to select 'Edit list of solvers' and move z3\_4\_4 to the top of the list and set the default prover to z3\_4\_4.
* Select ESC auto-checking on save and then save the file. This will give results of the ESC checker along with counter-examples if any errors occur.



* A model trace is also produced to show how the program can fail the assertion selected

TRACE of Q1\_2017.PairInsertionSort.sort(int,int)

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:9: requires 0 < l && l <= r && r < a.length;

VALUE: 0 === 0

VALUE: l === 9013

VALUE: 0 < l === true

VALUE: l === 9013

VALUE: r === 9076

VALUE: l <= r === true

VALUE: 0 < l && l <= r === true

VALUE: r === 9076

VALUE: a === REF!val!8

VALUE: a.length === 9078

VALUE: r < a.length === true

VALUE: 0 < l && l <= r && r < a.length === true

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:9: UndefinedNullDeReference assertion: !(THIS.a != null) || THIS.a != null

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:14: int left = l

VALUE: l === 9013

VALUE: left === 9013

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:15: int right = r

VALUE: r === 9076

VALUE: right === 9076

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:23: int k = left

VALUE: left === 9013

VALUE: k === 9013

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:19: //@ loop\_invariant l <= k && k <= right;

VALUE: l === 9013

VALUE: k === 9013

VALUE: l <= k === true

VALUE: k === 9013

VALUE: right === 9076

VALUE: k <= right === true

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:19: LoopInvariantBeforeLoop assertion: \_JML\_\_conditionalResult\_26

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:20: //@ loop\_invariant l <= left && left <= right + 1 && right == r;

VALUE: l === 9013

VALUE: left === 9013

VALUE: l <= left === true

VALUE: left === 9013

VALUE: right === 9076

VALUE: 1 === 1

VALUE: right + 1 === 9077

VALUE: left <= right + 1 === true

VALUE: l <= left && left <= right + 1 === true

VALUE: right === 9076

VALUE: r === 9076

VALUE: right == r === true

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:20: LoopInvariantBeforeLoop assertion: \_JML\_\_conditionalResult\_35

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:21: //@ loop\_invariant (\forall int i; l <= i && i < left; a[i] <= a[i + 1]);

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:21: LoopInvariantBeforeLoop assertion: (\forall int i; l <= i && i < left; THIS.a[i] <= THIS.a[i + 1])

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:19: //@ loop\_invariant l <= k && k <= right;

VALUE: l === 9013

VALUE: k === 9073

VALUE: l <= k === true

VALUE: k === 9073

VALUE: right === 9076

VALUE: k <= right === true

VALUE: l <= k && k <= right === true

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:20: //@ loop\_invariant l <= left && left <= right + 1 && right == r;

VALUE: l === 9013

VALUE: left === 9076

VALUE: l <= left === true

VALUE: left === 9076

VALUE: right === 9076

VALUE: 1 === 1

VALUE: right + 1 === 9077

VALUE: left <= right + 1 === true

VALUE: l <= left && left <= right + 1 === true

VALUE: right === 9076

VALUE: r === 9076

VALUE: right == r === true

VALUE: l <= left && left <= right + 1 && right == r === true

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:21: //@ loop\_invariant (\forall int i; l <= i && i < left; a[i] <= a[i + 1]);

VALUE: \forall int i; l <= i && i < left; a[i] <= a[i + 1] === (error "line 1533 column 233: invalid get-value term, term must be ground and must not contain quantifiers")

VALUE: (\forall int i; l <= i && i < left; a[i] <= a[i + 1]) === (error "line 1484 column 233: invalid get-value term, term must be ground and must not contain quantifiers")

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:18: //@ decreases right + 1 - left;

VALUE: right === 9076

VALUE: 1 === 1

VALUE: right + 1 === 9077

VALUE: left === 9076

VALUE: right + 1 - left === 1

Initial value of Loop Decreases expression = 1

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:23: Loop test

VALUE: ++left === 9077

VALUE: right === 9076

VALUE: ++left <= right === false

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:54: int last = a[right]

VALUE: a === REF!val!8

VALUE: right === 9076

VALUE: a[right] === ( - 898 )

VALUE: last === ( - 898 )

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:54: PossiblyNullDeReference assertion: THIS.a != null

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:54: PossiblyNegativeIndex assertion: 0 <= right

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:54: PossiblyTooLargeIndex assertion: THIS.a.length > right

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:58: //@ loop\_invariant l <= right && right < r;

VALUE: l === 9013

VALUE: right === 9076

VALUE: l <= right === true

VALUE: right === 9076

VALUE: r === 9076

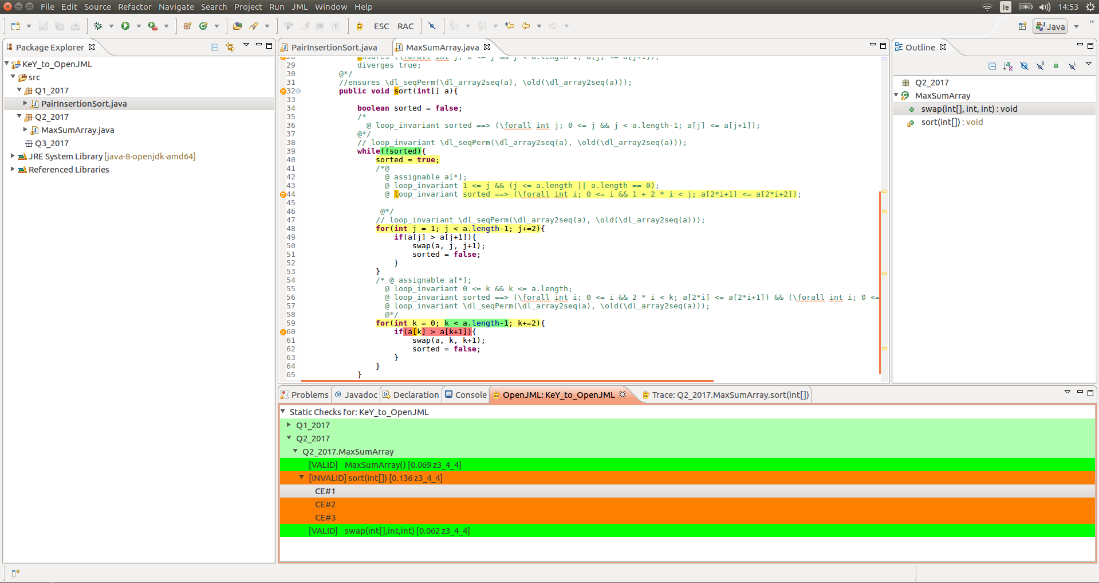
VALUE: right < r === false

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:58: LoopInvariantBeforeLoop assertion: \_JML\_\_conditionalResult\_190

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q1\_2017/PairInsertionSort.java:58: Invalid assertion (LoopInvariantBeforeLoop)

1. MaxSumArray

* **ERROR:**
  + Selecting the ESC button does not appear to be doing anything.
  + Have to set the OpenJML preferences to run ESC on every save
* The same procedure as set out for program 1 is required for all files for ESC checking.



TRACE of Q2\_2017.MaxSumArray.sort(int[])

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:32: requires a != null;

VALUE: a === REF!val!8

VALUE: null === NULL

VALUE: a != null === true

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:34: boolean sorted = false

VALUE: false === false

VALUE: sorted === false

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:39: Loop test

VALUE: sorted === false

VALUE: !sorted === true

VALUE: (!sorted) === true

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:40: sorted = true

VALUE: true === true

VALUE: sorted = true === true

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:48: int j = 1

VALUE: 1 === 1

VALUE: j === 1

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:43: //@ loop\_invariant 1 <= j && (j <= a.length || a.length == 0);

VALUE: 1 === 1

VALUE: j === 1

VALUE: 1 <= j === true

VALUE: j === 1

VALUE: a.length === 1798

VALUE: j <= a.length === true

VALUE: j <= a.length || a.length == 0 === true

VALUE: (j <= a.length || a.length == 0) === true

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:43: UndefinedNullDeReference assertion: !(a != null) || a != null

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:43: LoopInvariantBeforeLoop assertion: \_JML\_\_conditionalResult\_47

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:44: //@ loop\_invariant sorted ==> (\forall int i; 0 <= i && 1 + 2 \* i < j; a[2 \* i + 1] <= a[2 \* i + 2]);

VALUE: sorted === true

VALUE: \forall int i; 0 <= i && 1 + 2 \* i < j; a[2 \* i + 1] <= a[2 \* i + 2] === (error "line 1495 column 206: invalid get-value term, term must be ground and must not contain quantifiers")

VALUE: (\forall int i; 0 <= i && 1 + 2 \* i < j; a[2 \* i + 1] <= a[2 \* i + 2]) === (error "line 1405 column 206: invalid get-value term, term must be ground and must not contain quantifiers")

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:44: LoopInvariantBeforeLoop assertion: \_JML\_\_tmp54

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:43: //@ loop\_invariant 1 <= j && (j <= a.length || a.length == 0);

VALUE: 1 === 1

VALUE: j === 1797

VALUE: 1 <= j === true

VALUE: j === 1797

VALUE: a.length === 1798

VALUE: j <= a.length === true

VALUE: j <= a.length || a.length == 0 === true

VALUE: (j <= a.length || a.length == 0) === true

VALUE: 1 <= j && (j <= a.length || a.length == 0) === true

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:43: UndefinedNullDeReference assertion: !(a != null) || a != null

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:44: //@ loop\_invariant sorted ==> (\forall int i; 0 <= i && 1 + 2 \* i < j; a[2 \* i + 1] <= a[2 \* i + 2]);

VALUE: sorted === false

VALUE: sorted ==> (\forall int i; 0 <= i && 1 + 2 \* i < j; a[2 \* i + 1] <= a[2 \* i + 2]) === true

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:48: Loop test

VALUE: j === 1797

VALUE: a.length === 1798

VALUE: 1 === 1

VALUE: a.length - 1 === 1797

VALUE: j < a.length - 1 === false

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:48: PossiblyNullDeReference assertion: a != null

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:48: ArithmeticOperationRange assertion: !(0 < \_JML\_\_tmp79 && 1 < 0) || \_JML\_\_tmp79 <= 2147483647 + 1

VALUE: !(0 < \_JML\_\_tmp79 && 1 < 0) || \_JML\_\_tmp79 <= 2147483647 + 1 === true

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:48: ArithmeticOperationRange assertion: !(\_JML\_\_tmp79 < 0 && 0 < 1) || -2147483648 + 1 <= \_JML\_\_tmp79

VALUE: !(\_JML\_\_tmp79 < 0 && 0 < 1) || -2147483648 + 1 <= \_JML\_\_tmp79 === true

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:59: int k = 0

VALUE: 0 === 0

VALUE: k === 0

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:59: Loop test

VALUE: k === ( - 1 )

VALUE: a.length === 1798

VALUE: 1 === 1

VALUE: a.length - 1 === 1797

VALUE: k < a.length - 1 === true

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:59: PossiblyNullDeReference assertion: a != null

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:59: ArithmeticOperationRange assertion: !(0 < \_JML\_\_tmp163 && 1 < 0) || \_JML\_\_tmp163 <= 2147483647 + 1

VALUE: !(0 < \_JML\_\_tmp163 && 1 < 0) || \_JML\_\_tmp163 <= 2147483647 + 1 === true

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:59: ArithmeticOperationRange assertion: !(\_JML\_\_tmp163 < 0 && 0 < 1) || -2147483648 + 1 <= \_JML\_\_tmp163

VALUE: !(\_JML\_\_tmp163 < 0 && 0 < 1) || -2147483648 + 1 <= \_JML\_\_tmp163 === true

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:60: if (a[k] > a[k + 1]) ...

VALUE: a === REF!val!8

VALUE: k === ( - 1 )

VALUE: a[k] === 29

VALUE: a === REF!val!8

VALUE: k === ( - 1 )

VALUE: 1 === 1

VALUE: k + 1 === 0

VALUE: a[k + 1] === 29

VALUE: a[k] > a[k + 1] === false

VALUE: (a[k] > a[k + 1]) === false

Condition = false

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:60: PossiblyNullDeReference assertion: a != null

/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:60: PossiblyNegativeIndex assertion: 0 <= k

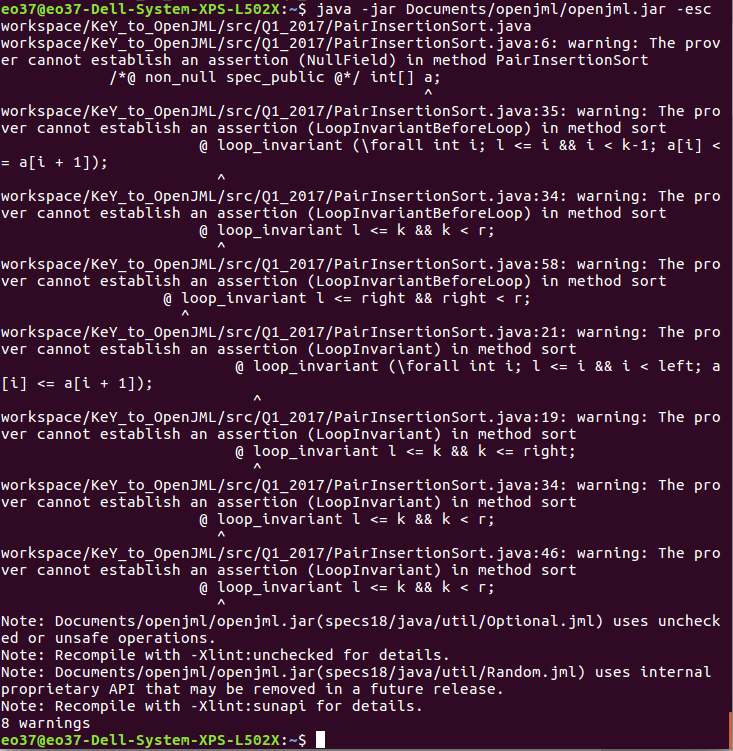
/home/eo37/workspace/KeY\_to\_OpenJML/src/Q2\_2017/MaxSumArray.java:60: Invalid assertion (PossiblyNegativeIndex)

Solving CounterExamples

Command Line

* - esc

1. PairInsertionSort
   * Static Checking errors are also found in PairInsertionSort
   * Requires matching with Eclipse plugin errors



1. MaxSumArray
   * Static Checking errors are also found in MaxSumArray
   * Requires matching with Eclipse plugin errors

