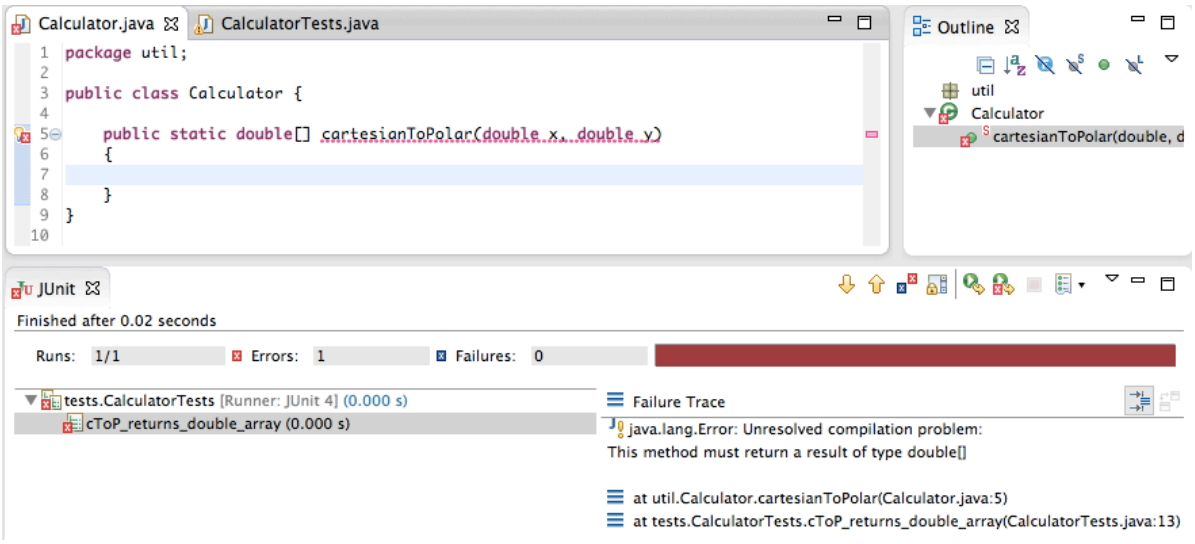
ECSE 428 Assignment B

Amee Joshipura (260461226)
Lilly Tong (260459522)

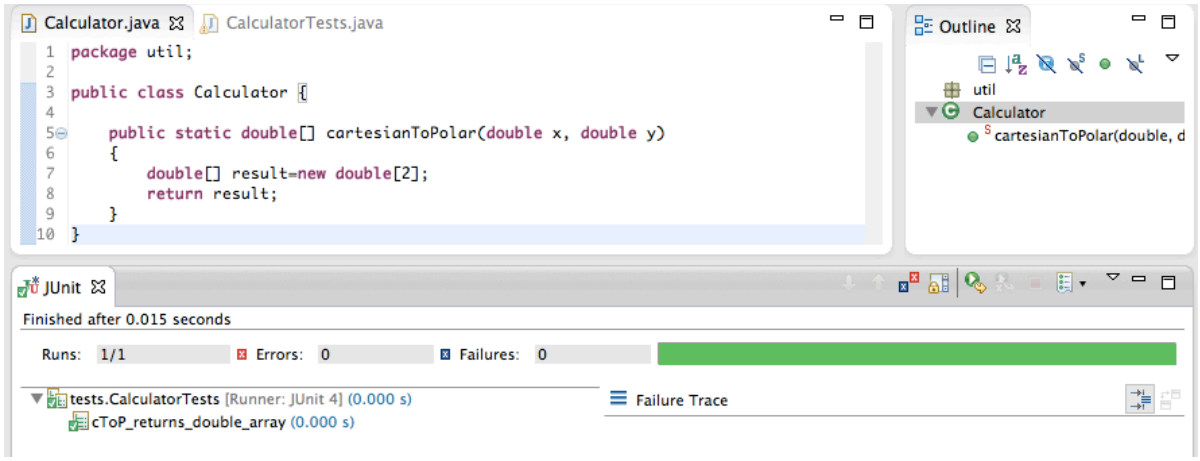
JUnit Test Cases & Screen shots

Test 1
Test name: cToP_returns_double_array
Call setup: cartesianToPolar(3,4)
Expected: cartesianToPolar() returns a double array

Fail:

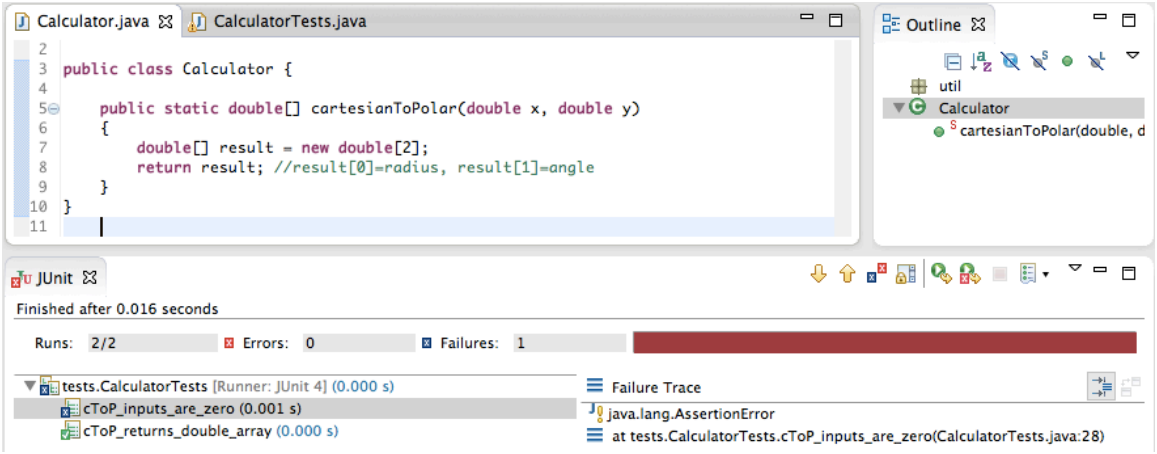


Succeed:

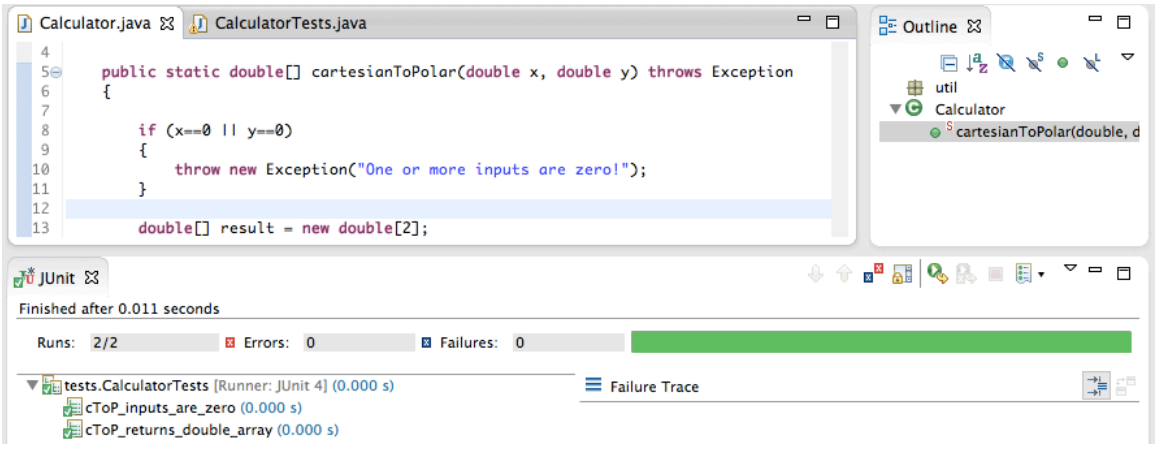


Test 2
Test name: cToP_inputs_are_zero
Call setup: cartesianToPolar(0,0)
Expected: cartesianToPolar() throws exception with message “One or more inputs are zero!”

Fail:

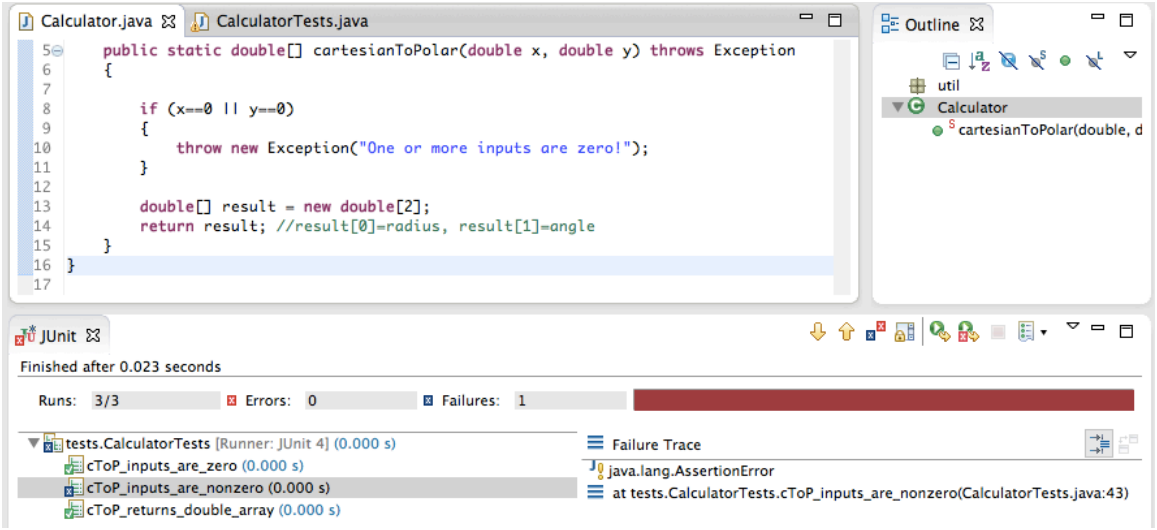


Succeed:

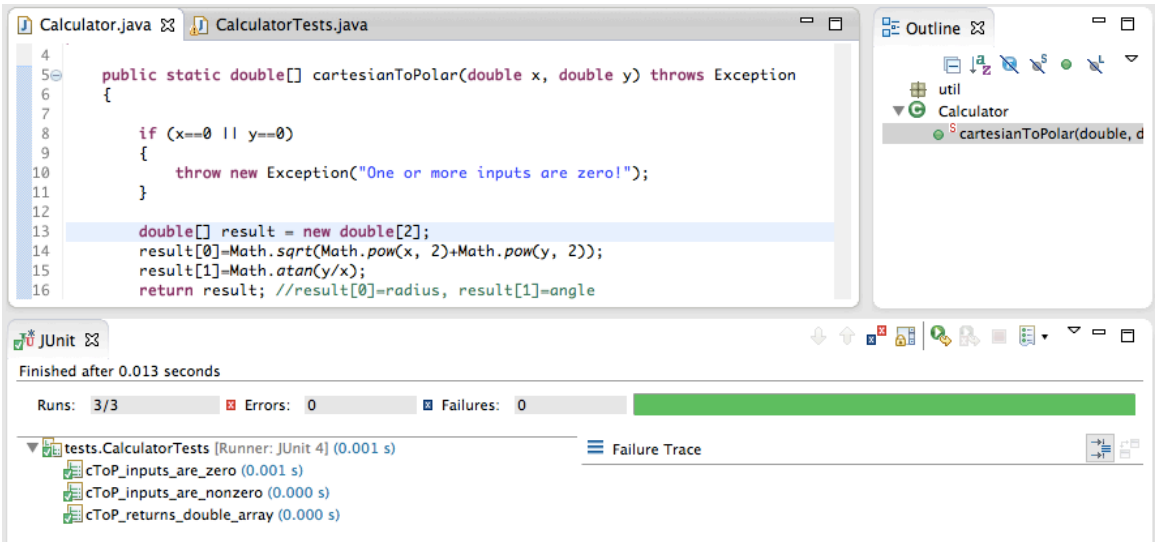


Test 3
Test name: cToP_inputs_are_nonzero
Call setup: cartesianToPolar(3,4)
Expected: cartesianToPolar() returns [5, arctan(4.0/3.0)]

Fail:

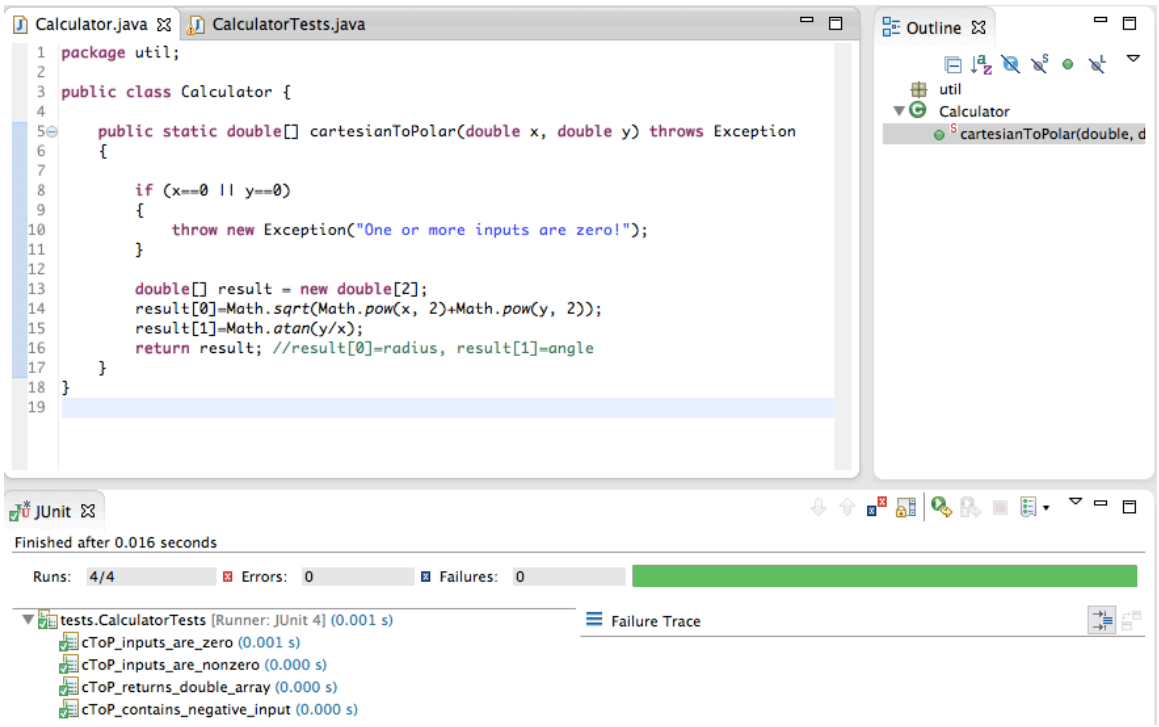


Succeed:



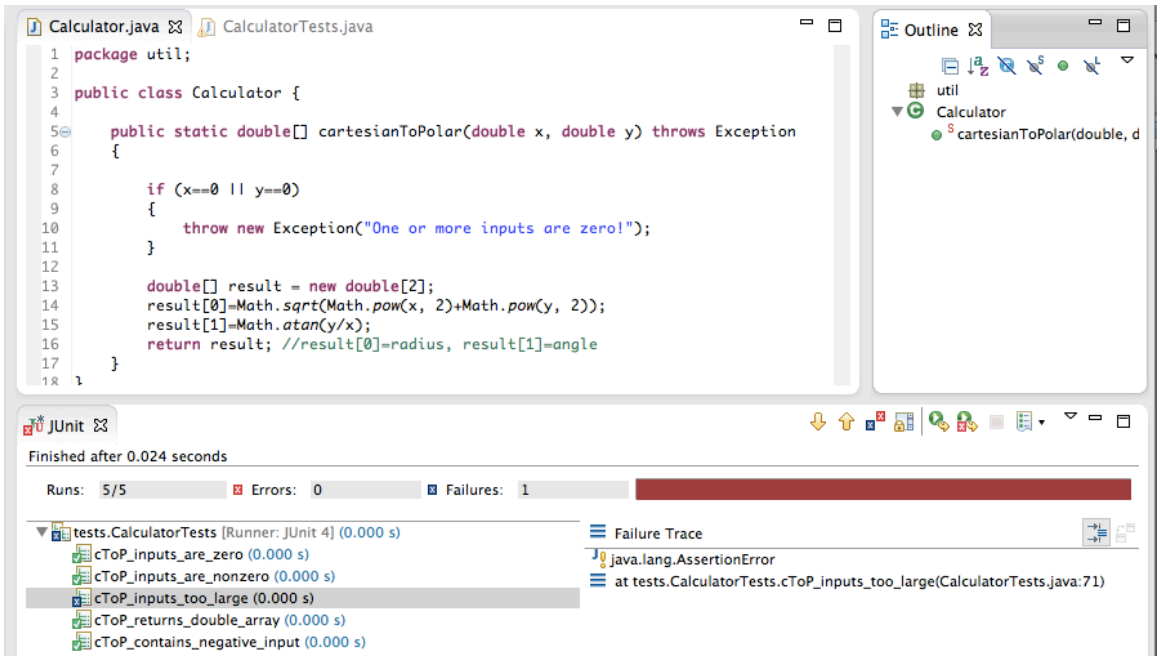
Test 4
Test name: cToP_contains_negative_input
Call setup: cartesianToPolar(3,-4)
Expected: cartesianToPolar() returns [5, arctan(-4.0/3.0)]

Succeed:

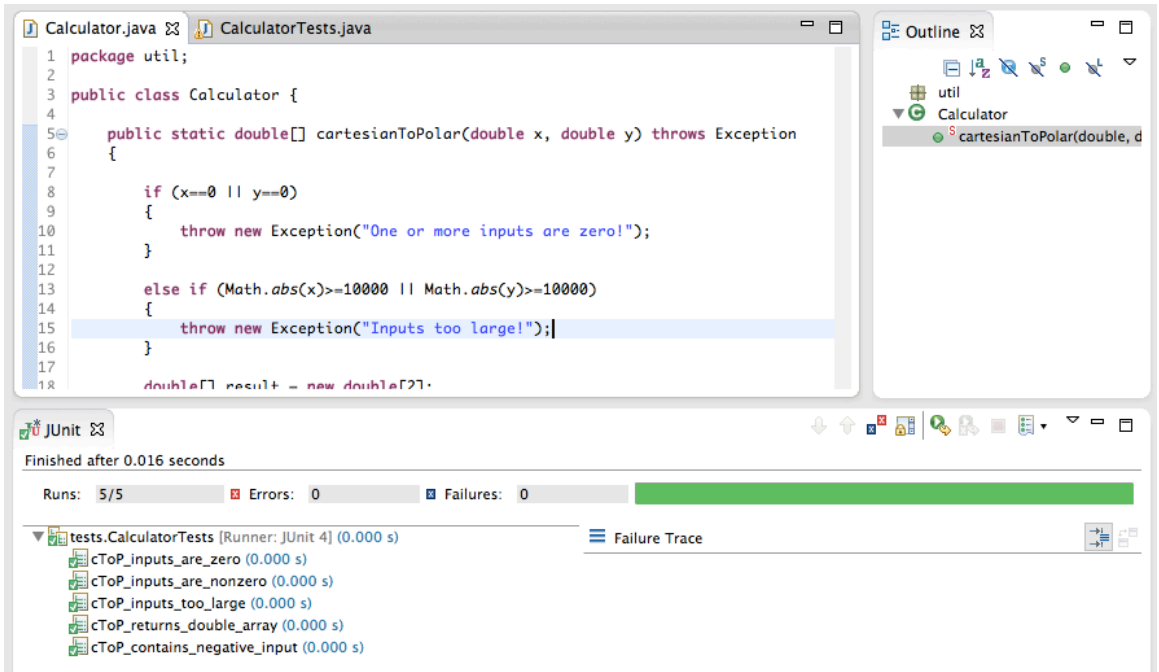


Test 5
Test name: cToP_inputs_too_large
Call setup: cartesianToPolar(10000,-10000)
Expected: cartesianToPolar() throws exception with message “Inputs too large!”

Fail:

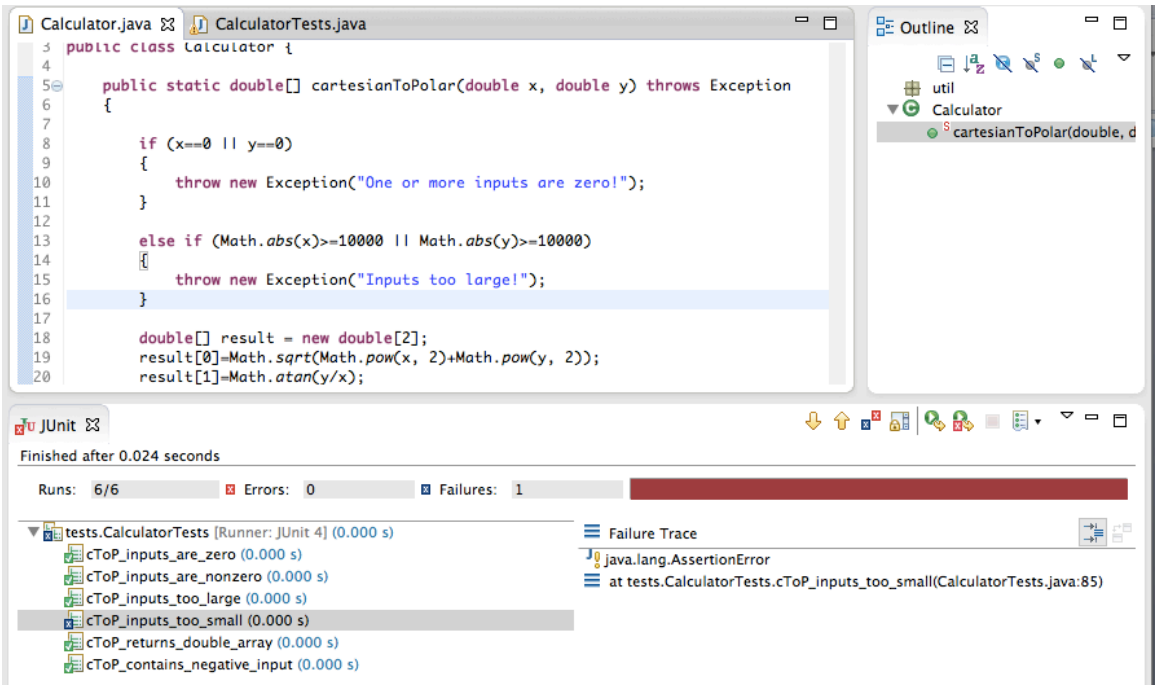


Succeed:

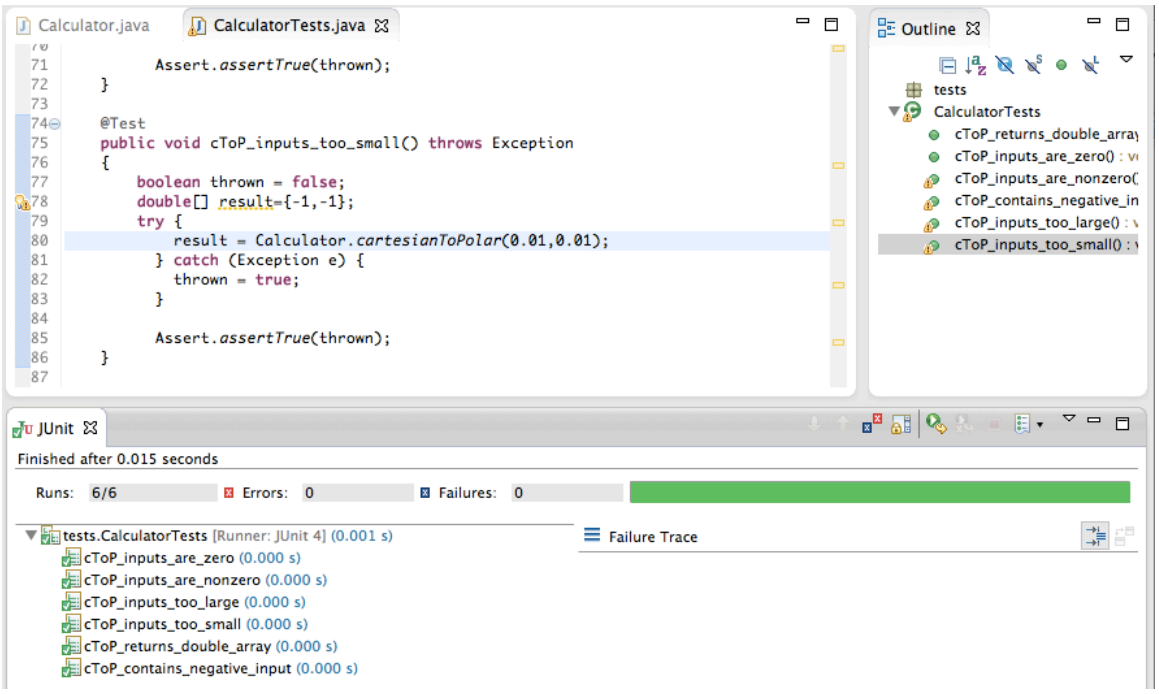


Test 6
Test name: cToP_inputs_too_small
Call setup: cartesianToPolar(0.01,-0.01)
Expected: cartesianToPolar() throws exception with message “Inputs too small!”

Fail:

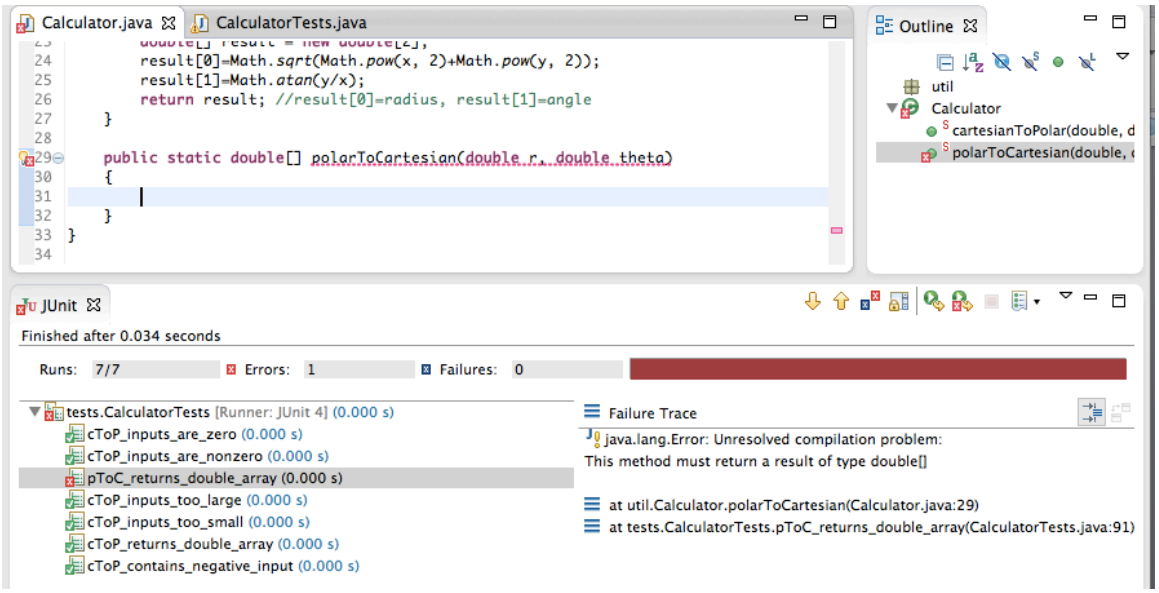


Succeed:

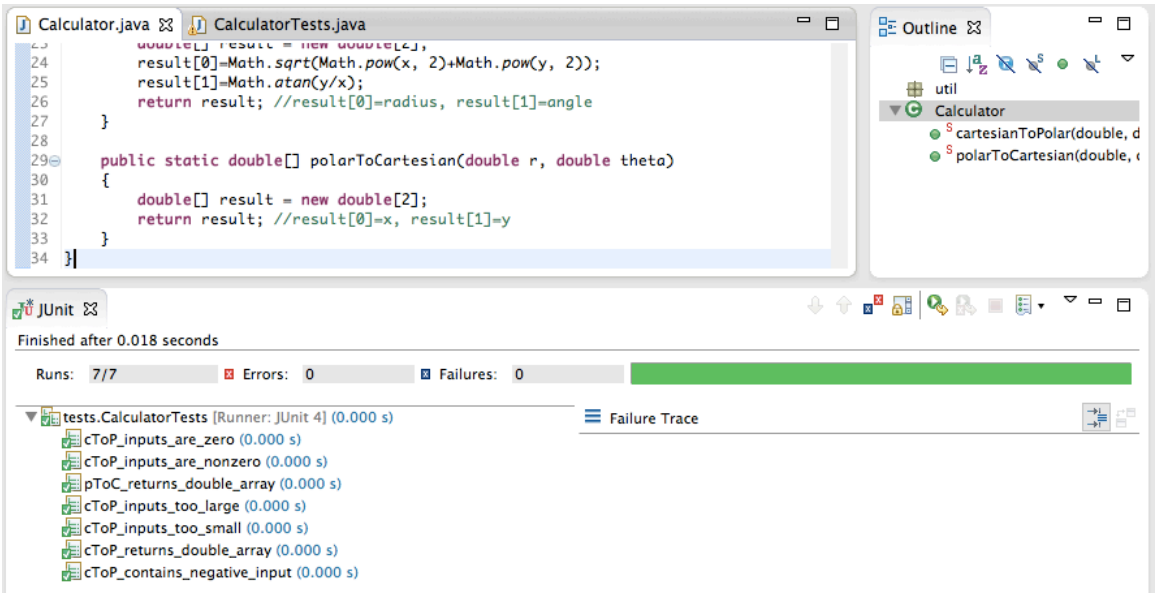


Test 7
Test name: pToC_returns_double_array
Call setup: polarToCartesian(3,0.5)
Expected: polarToCartesian() returns a double array

Fail:

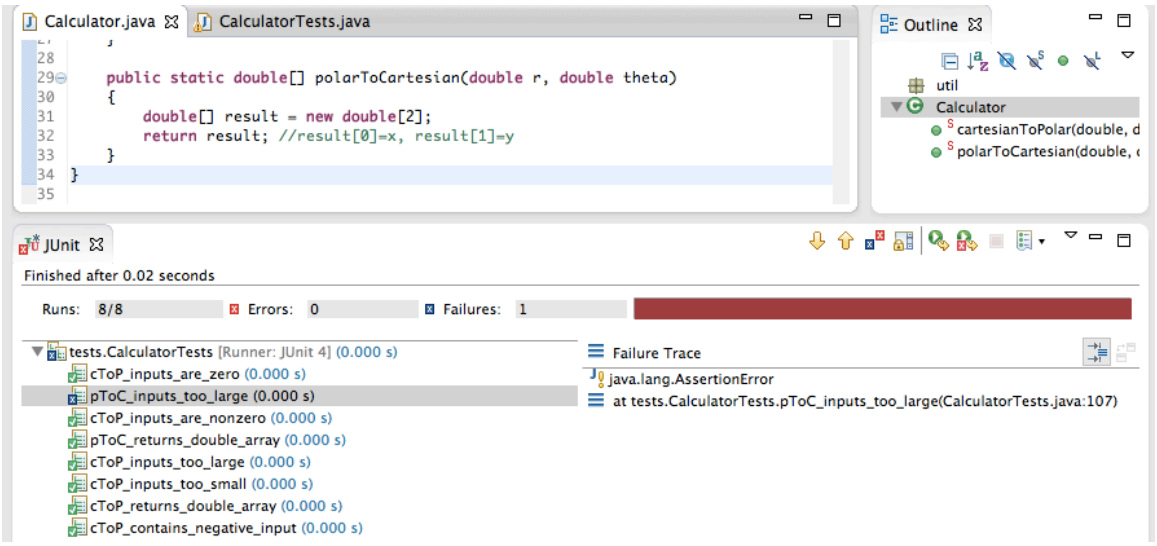


Succeed:

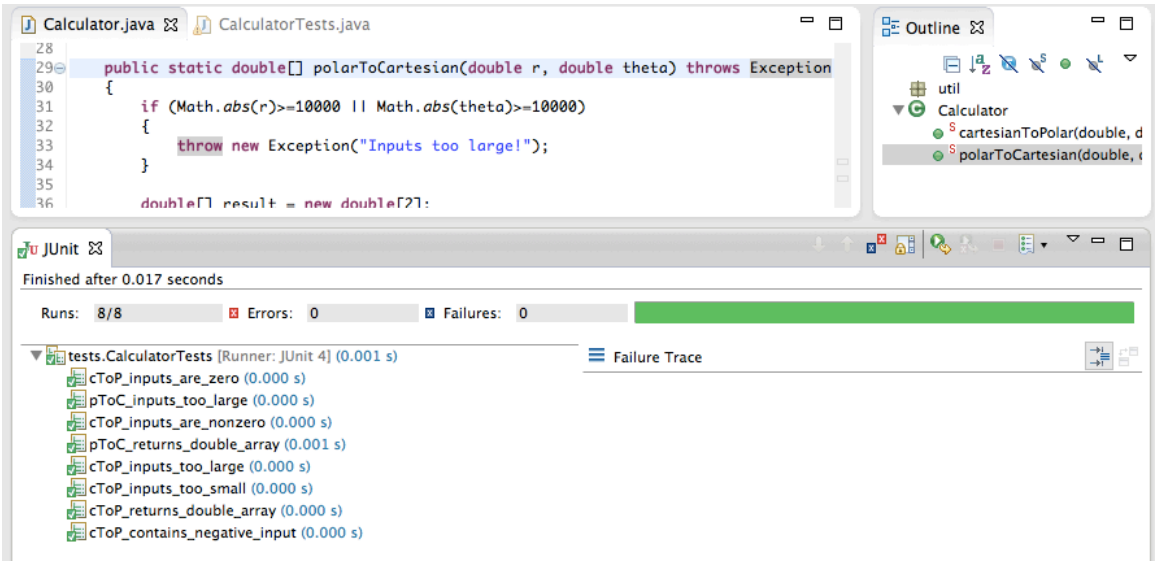


Test 8
Test name: pToC_inputs_too_large
Call setup: polarToCartesian(10000,-10000)
Expected: polarToCartesian () throws exception with message “Inputs too large!”

Fail:

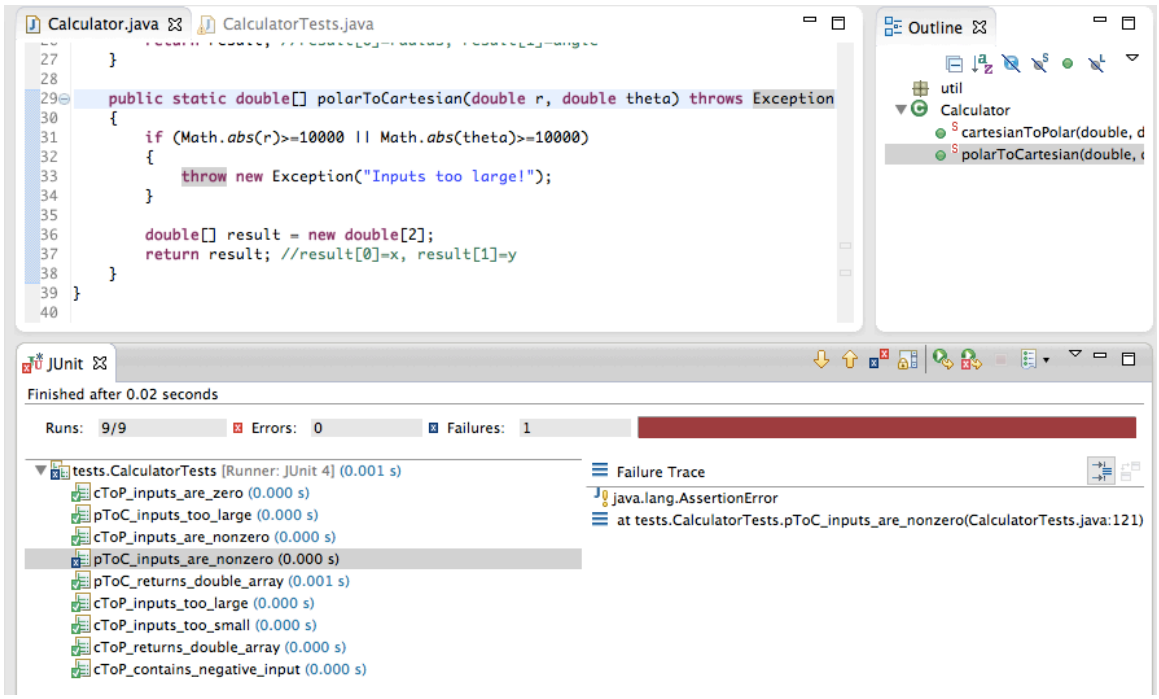


Succeed:

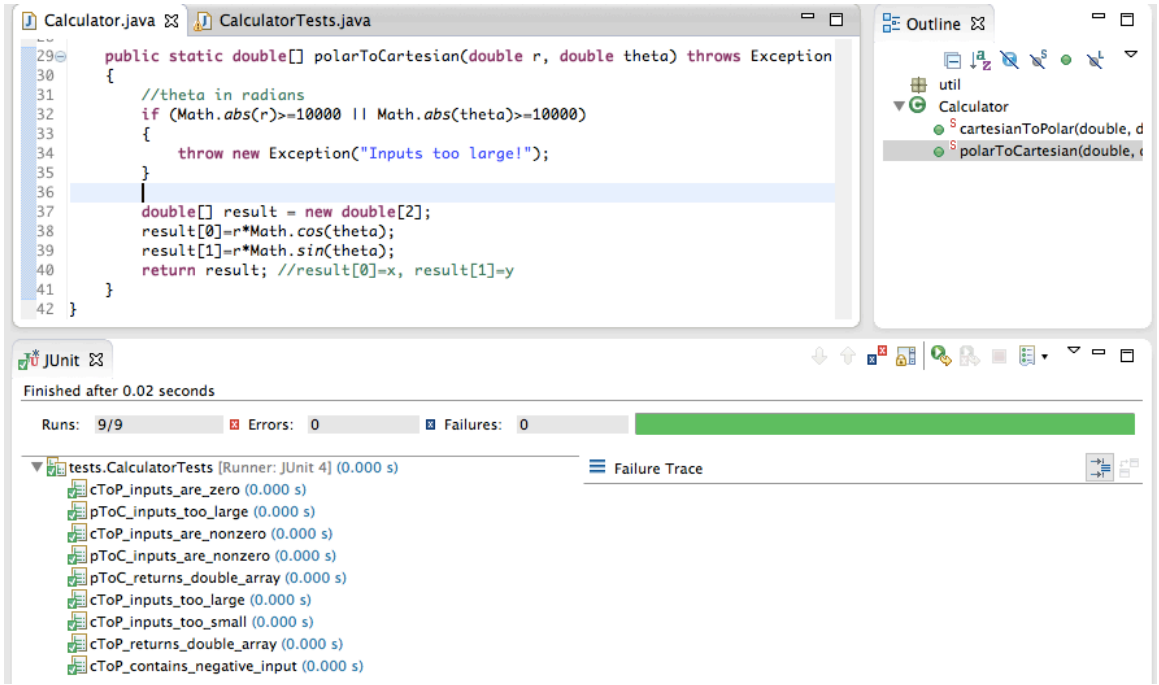


Test 9
Test name: pToC_inputs_are_nonzero
Call setup: polarToCartesian(5,arctan(4/3))
Expected: polarToCartesian() returns [3,4]

Fail:

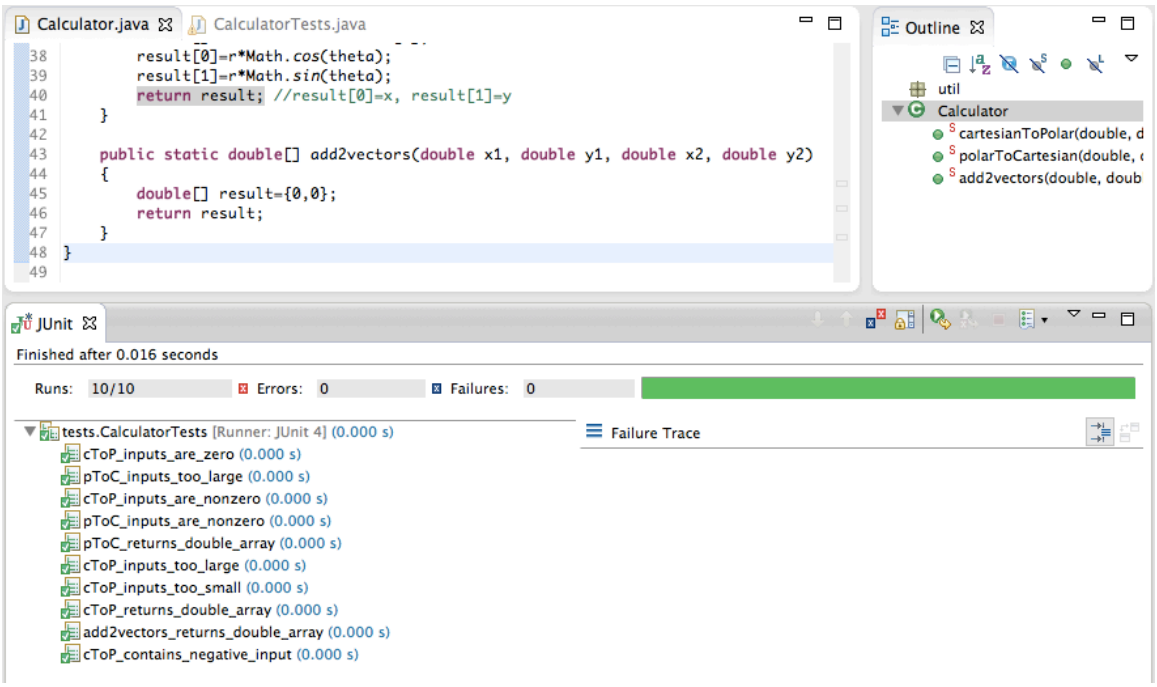


Succeed:



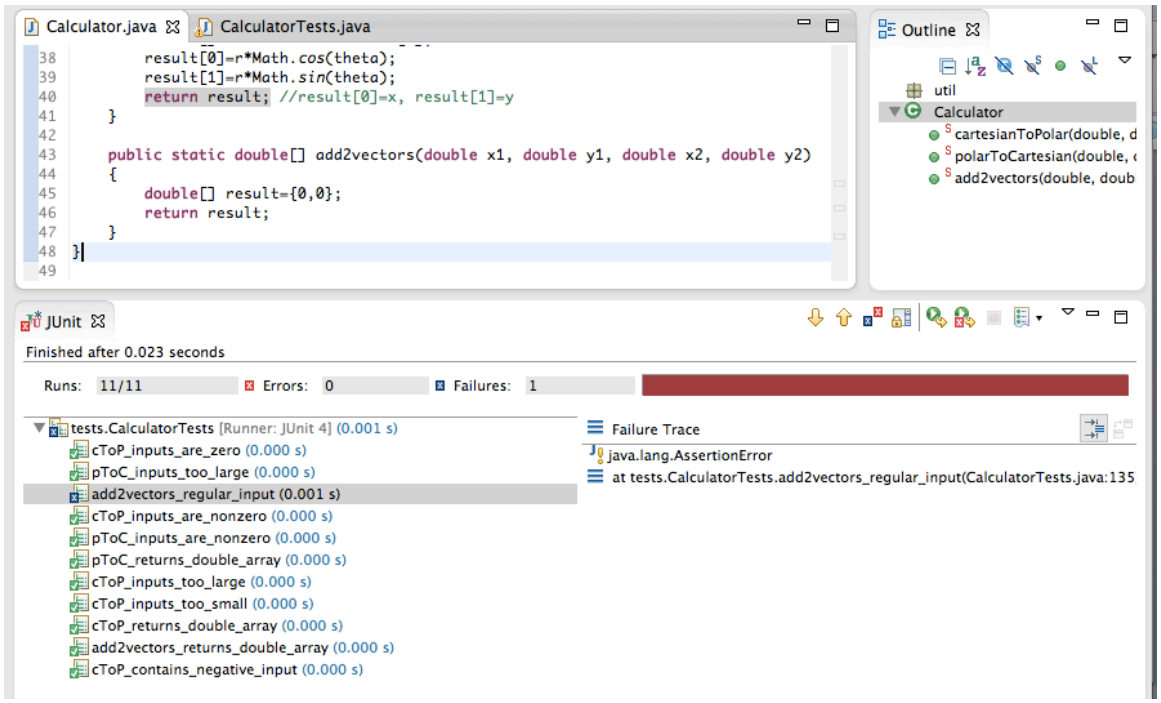
Test 10
Test name: add2vectors_returns_double_array
Call setup: add2vectors(3,4,5,6)
Expected: add2vectors() returns a double array

Succeed:

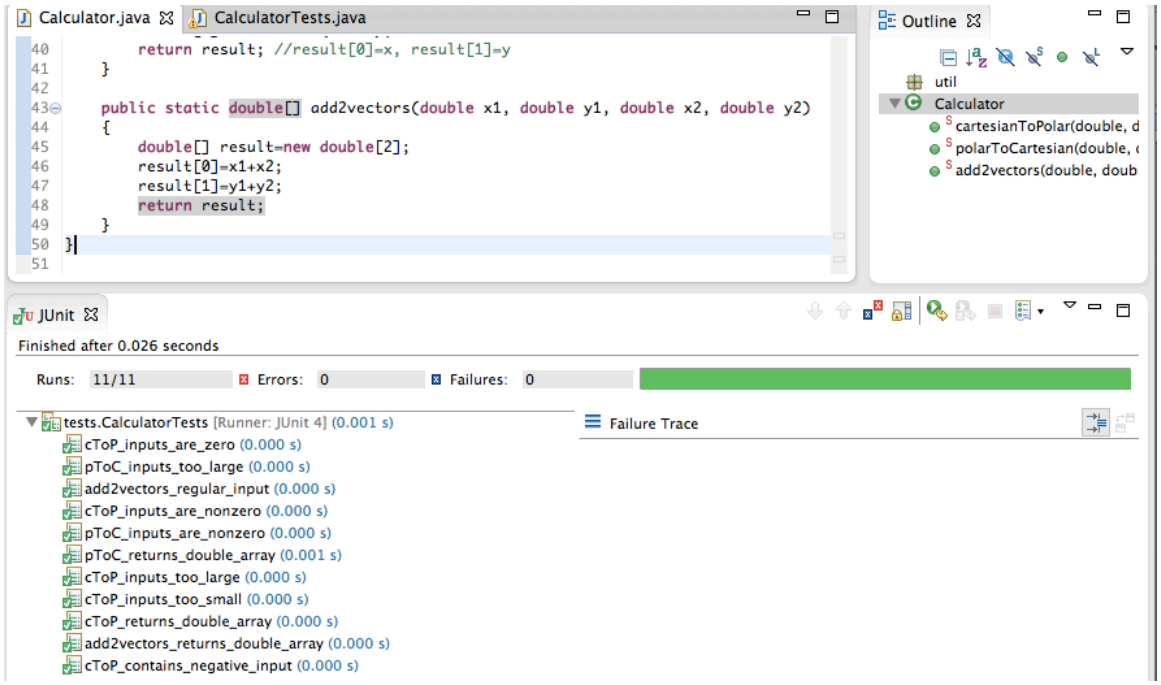


Test 11
Test name: add2vectors_regular_input
Call setup: add2vectors(3,4,5,6)
Expected: add2vectors() returns [8,10]

Fail:

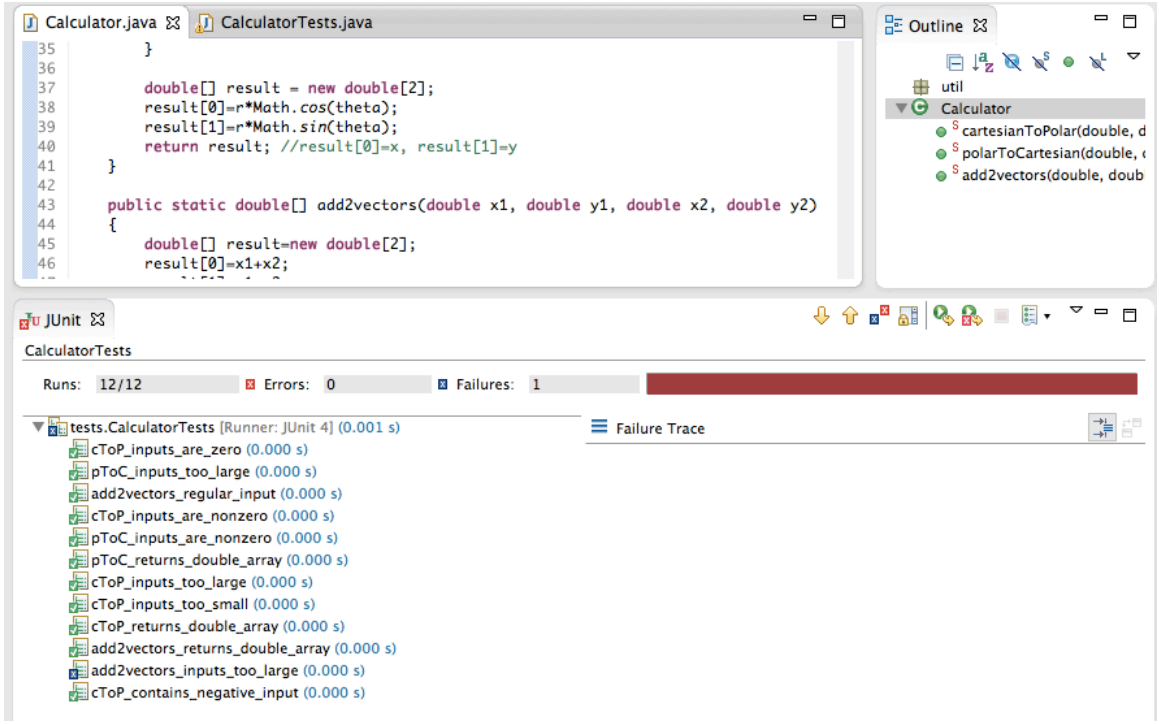


Succeed:

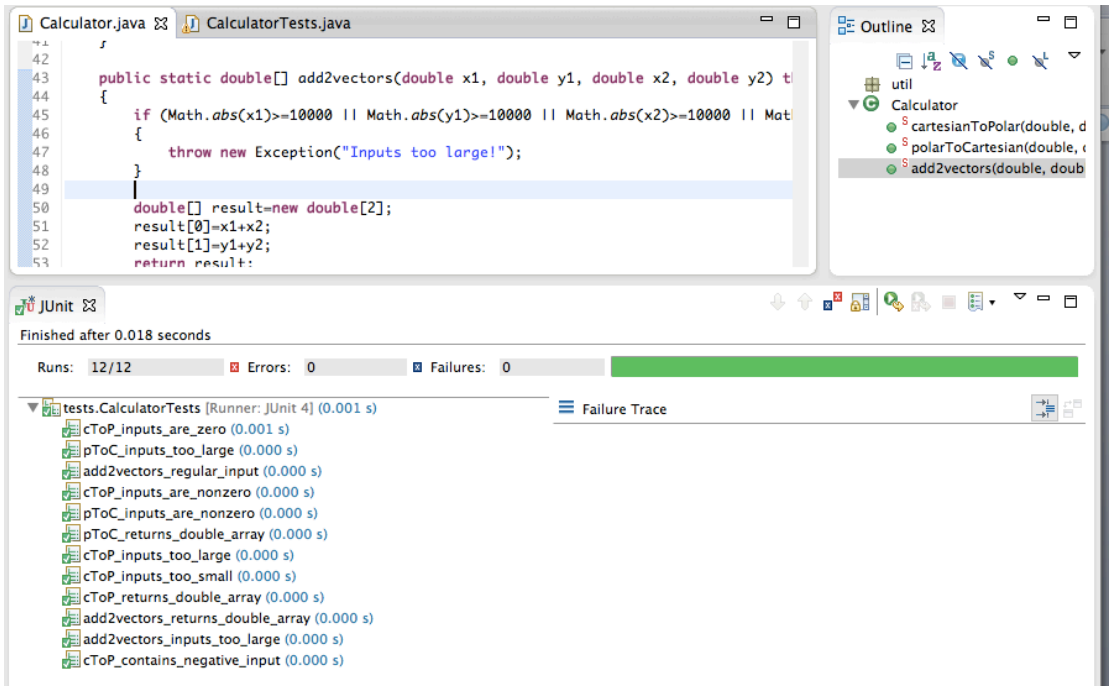


Test 12
Test name: add2vectors_inputs_too_large
Call setup: add2vectors(10000,-10000,10000,-10000)
Expected: add2vectors() throws exception with message “Inputs too large!”

Fail:

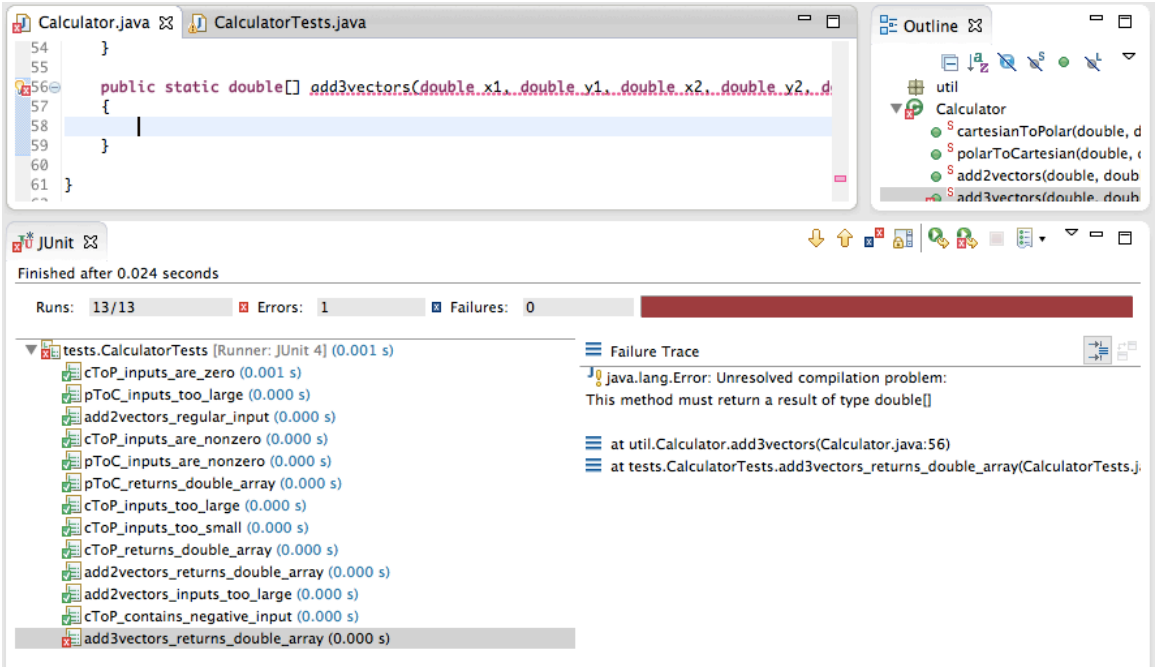


Succeed:

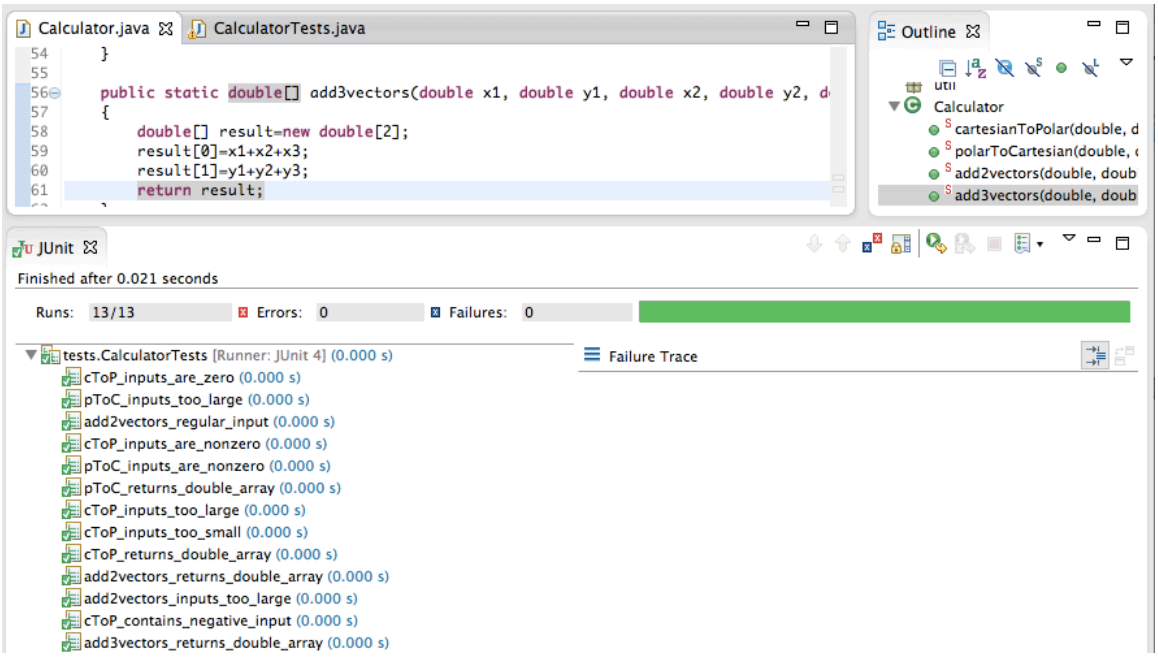


Test 13
Test name: add3vectors_returns_double_array
Call setup: add3vectors(3,4,5,6,7,8)
Expected: add3vectors() returns a double array

Fail:

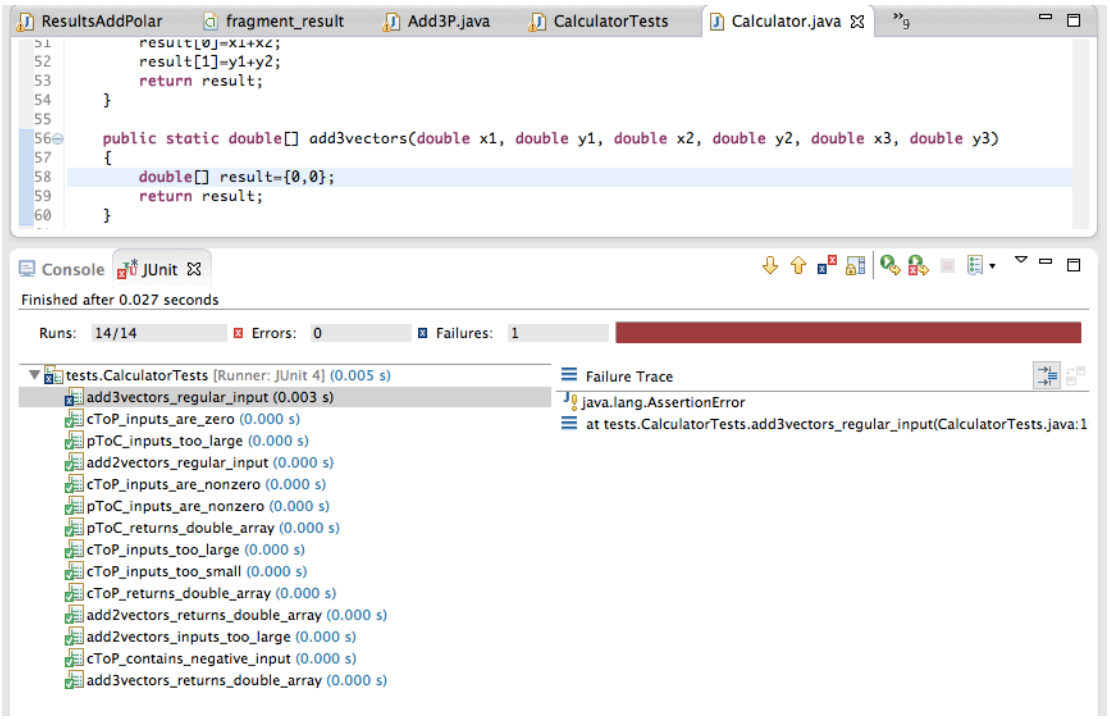


Succeed:

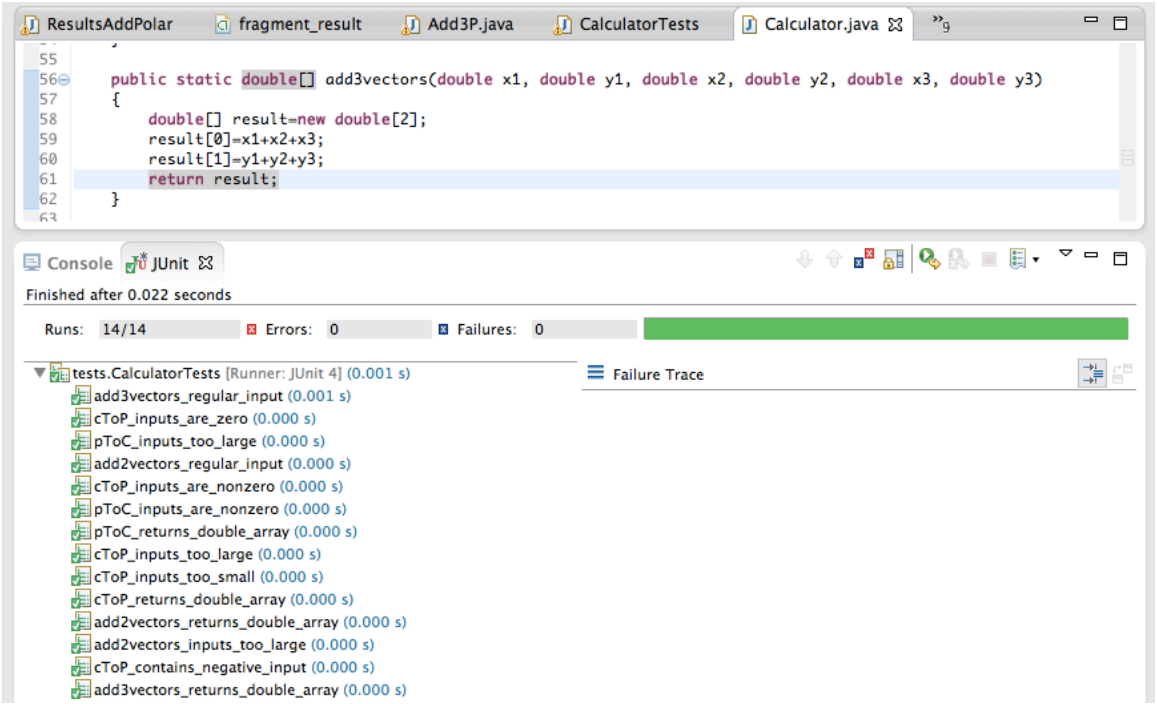


Test 14
Test name: add3vectors_regular_input
Call setup: add3vectors(3,4,5,6,7,8)
Expected: add2vectors() returns [15,18]

Fail:

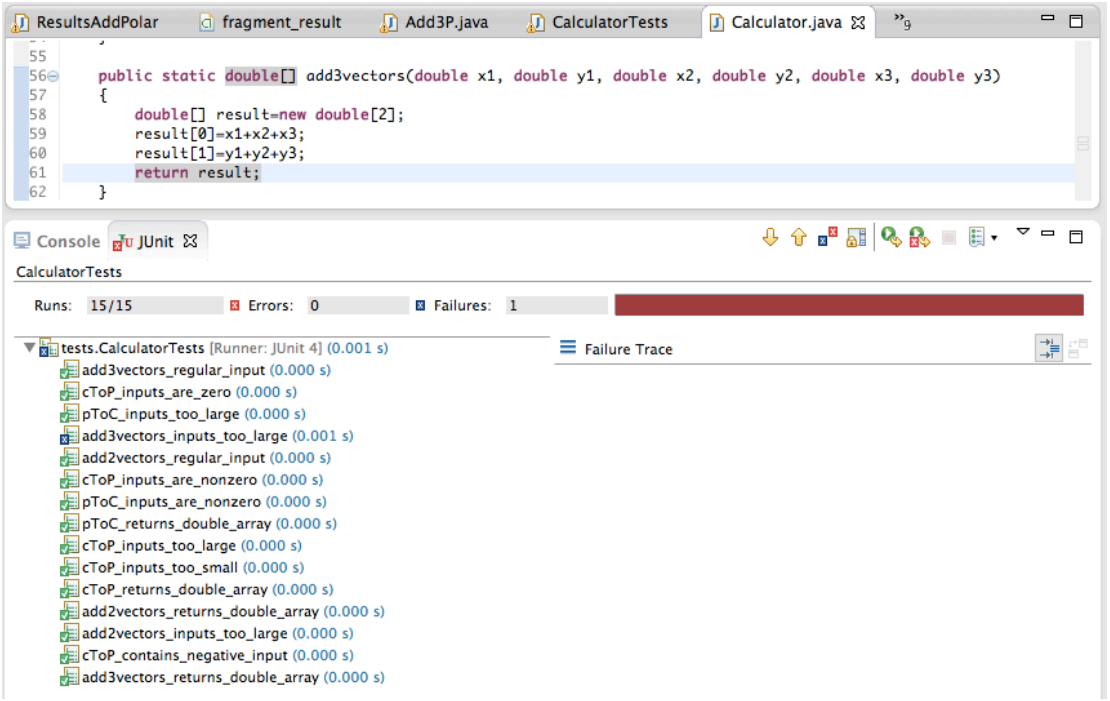


Succeed:

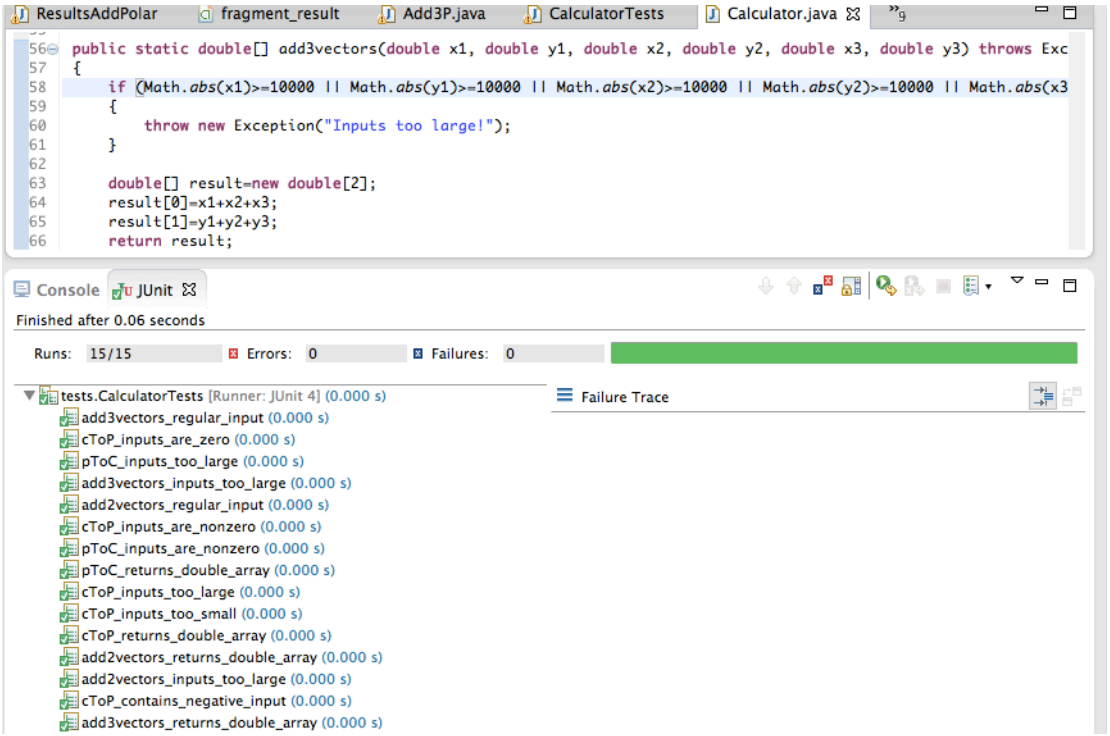


Test 15
Test name: add3vectors_inputs_too_large
Call setup: add3vectors(10000,-10000,10000,-10000,10000,-10000)
Expected: add3vectors() throws exception with message “Inputs too large!”

Fail:

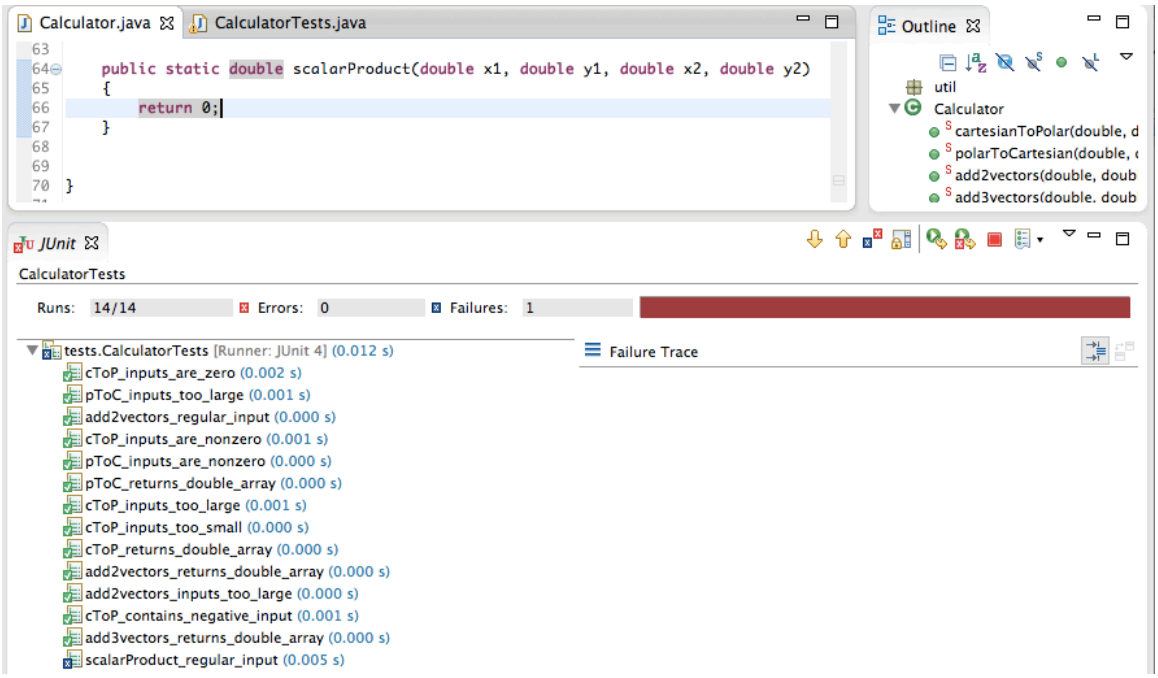


Succeed:

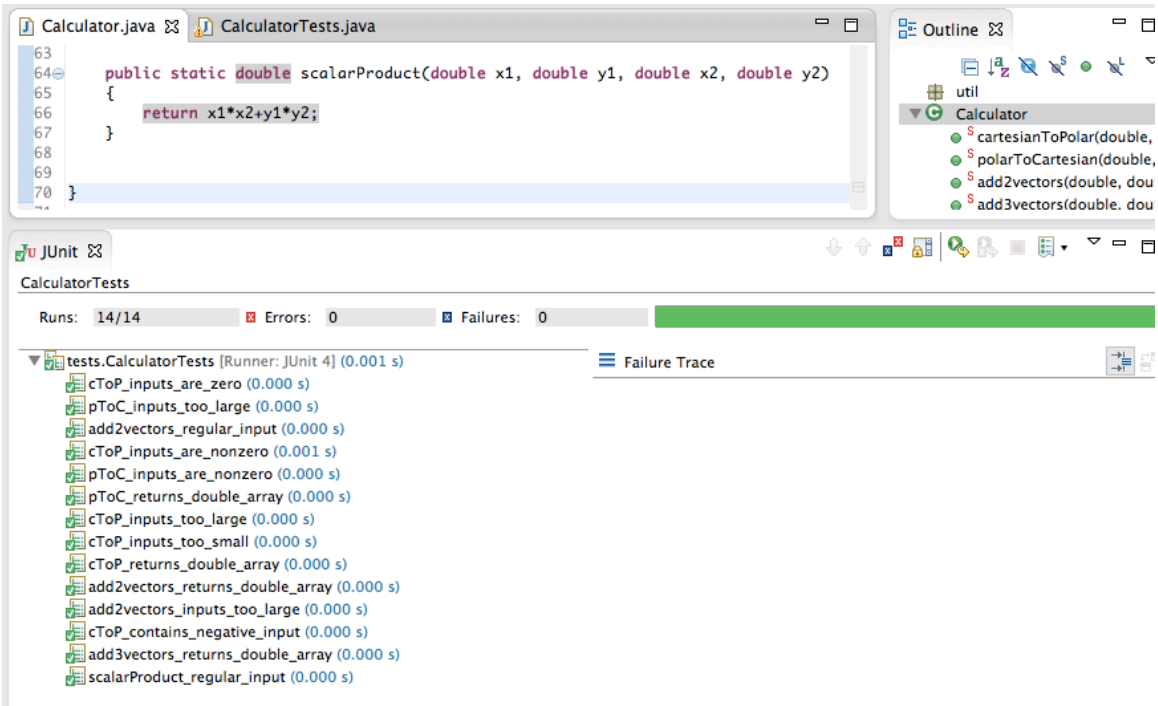


Test 16
Test name: scalarProduct_regular_input
Call setup: scalarProduct(3,4,5,6)
Expected: scalarProduct() returns a 39

Fail:

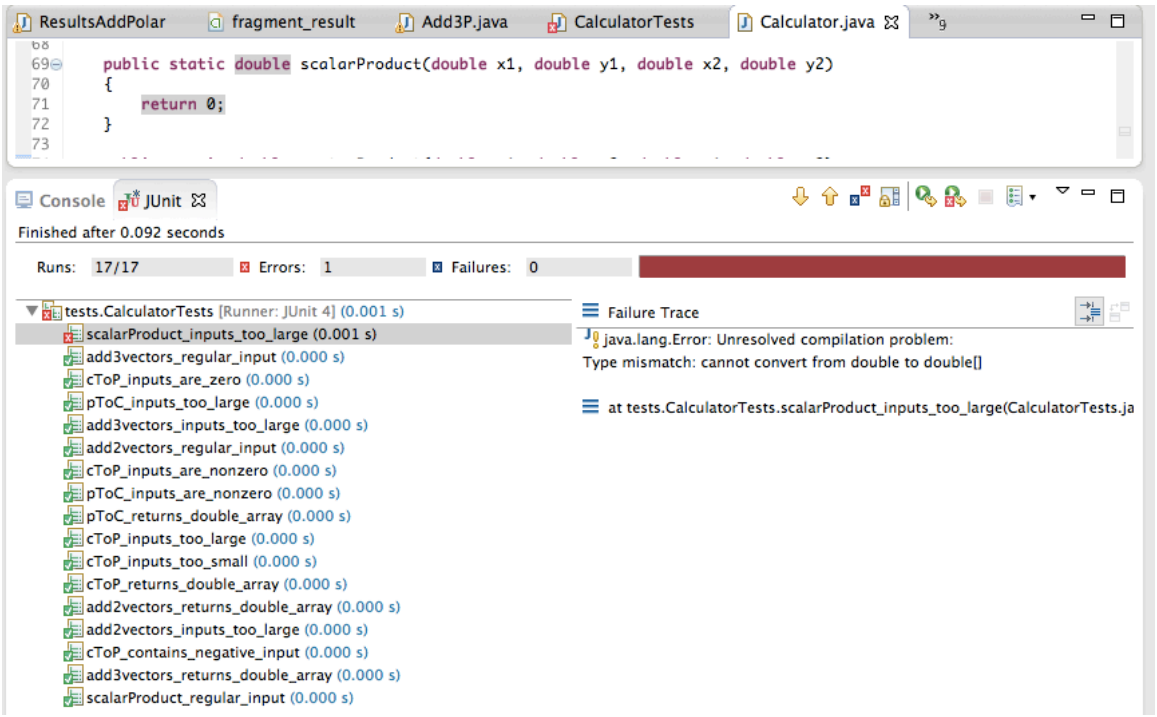


Succeed:

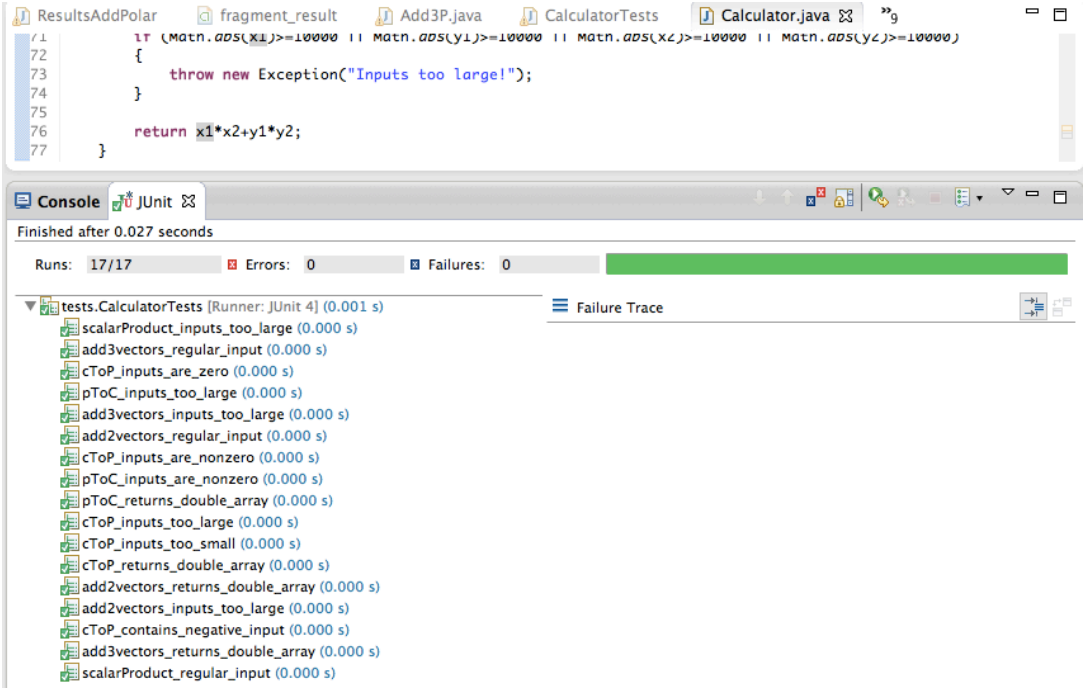


Test 17
Test name: scalarProduct_inputs_too_large
Call setup: scalarProduct(10000,-10000,10000,-10000)
Expected: scalarProduct() throws exception with message “Inputs too large!”

Fail:

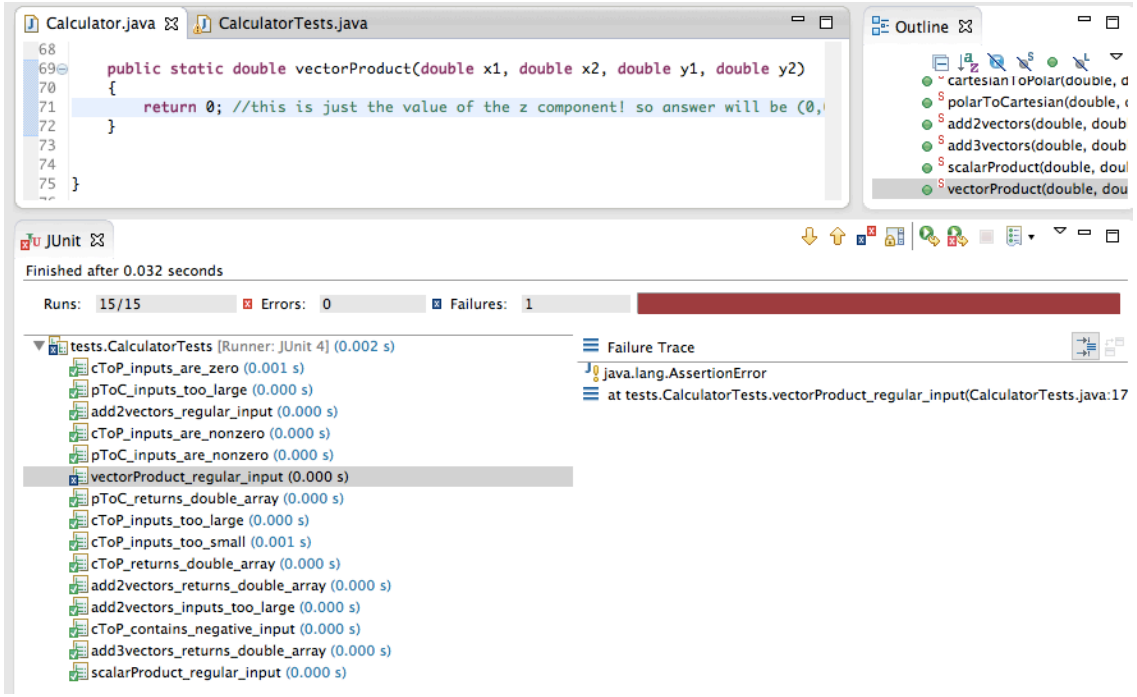


Succeed:

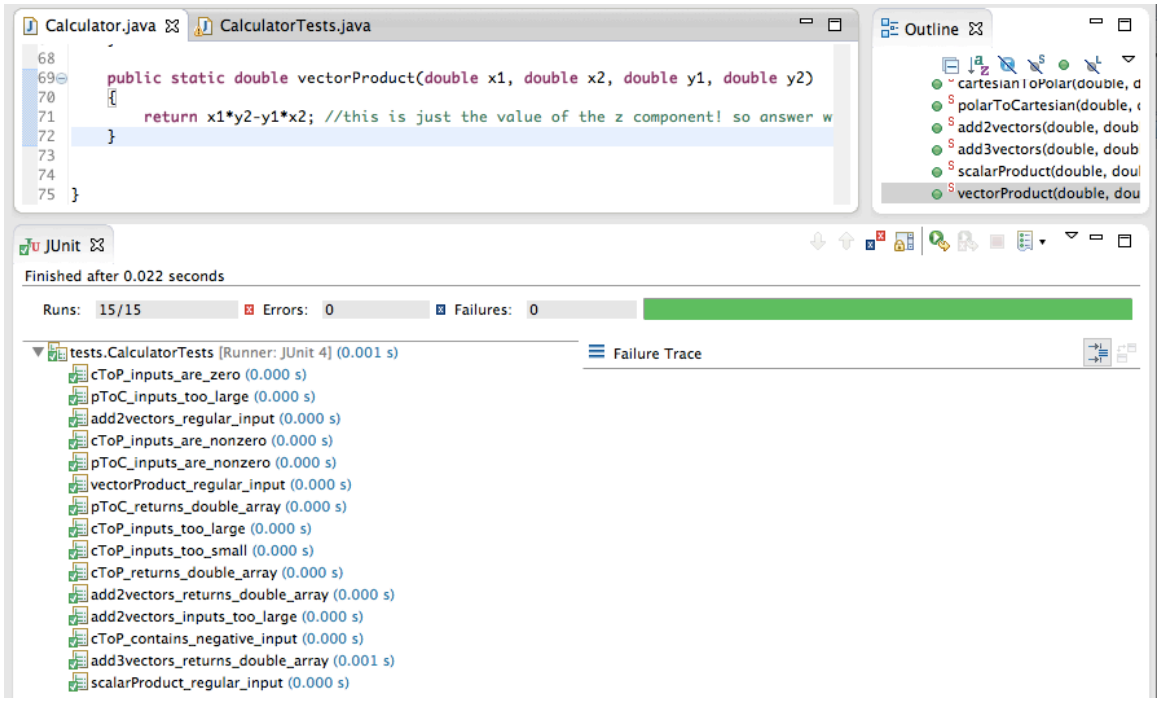


Test 18
Test name: vectorProduct_regular_input
Call setup: vectorProduct(3,4,5,6)
Expected: vectorProduct() returns a -2

Fail:

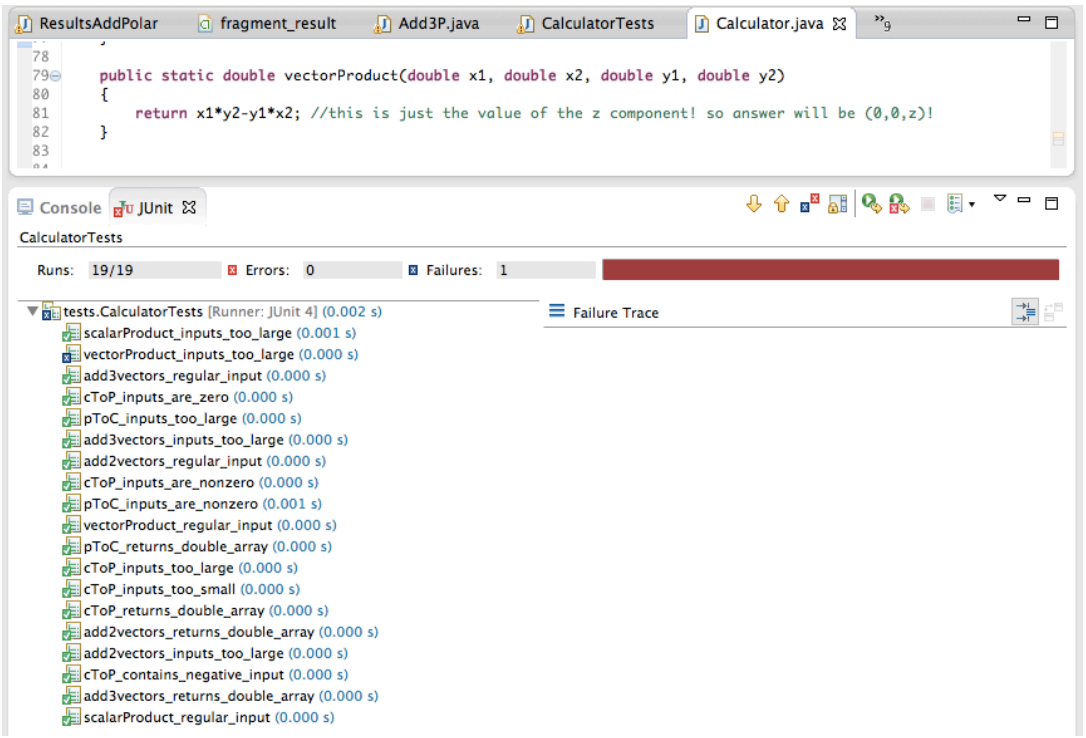


Succeed:



Test 19
Test name: vectorProduct_inputs_too_large
Call setup: vectorProduct(10000,-10000,10000,-10000)
Expected: vectorProduct() throws exception with message “Inputs too large!”

Fail:



Succeed: