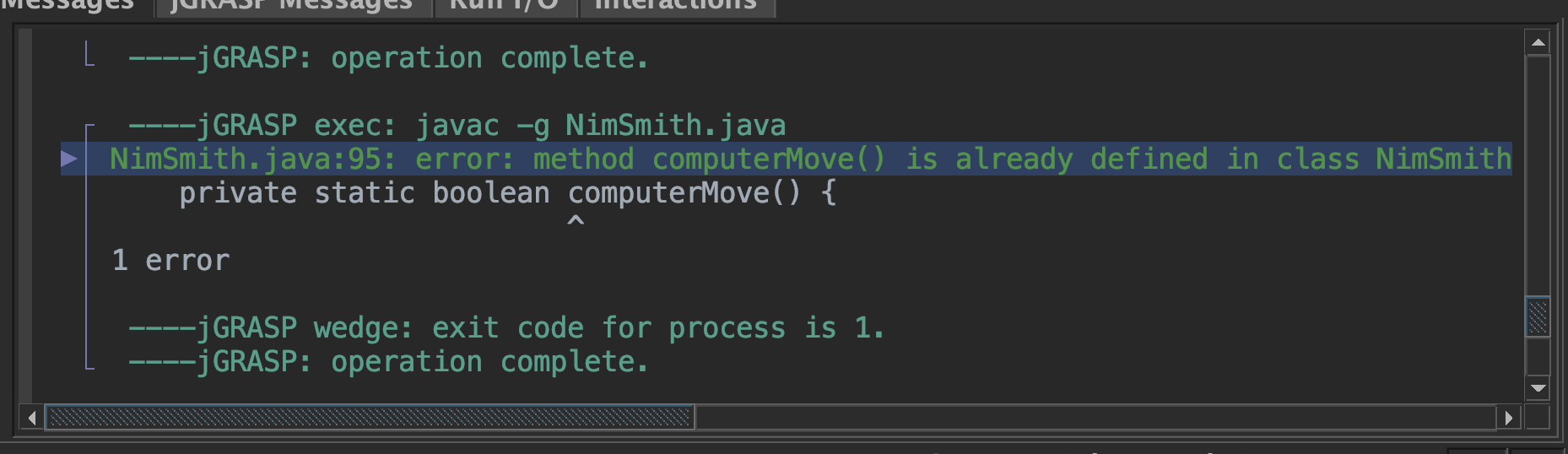
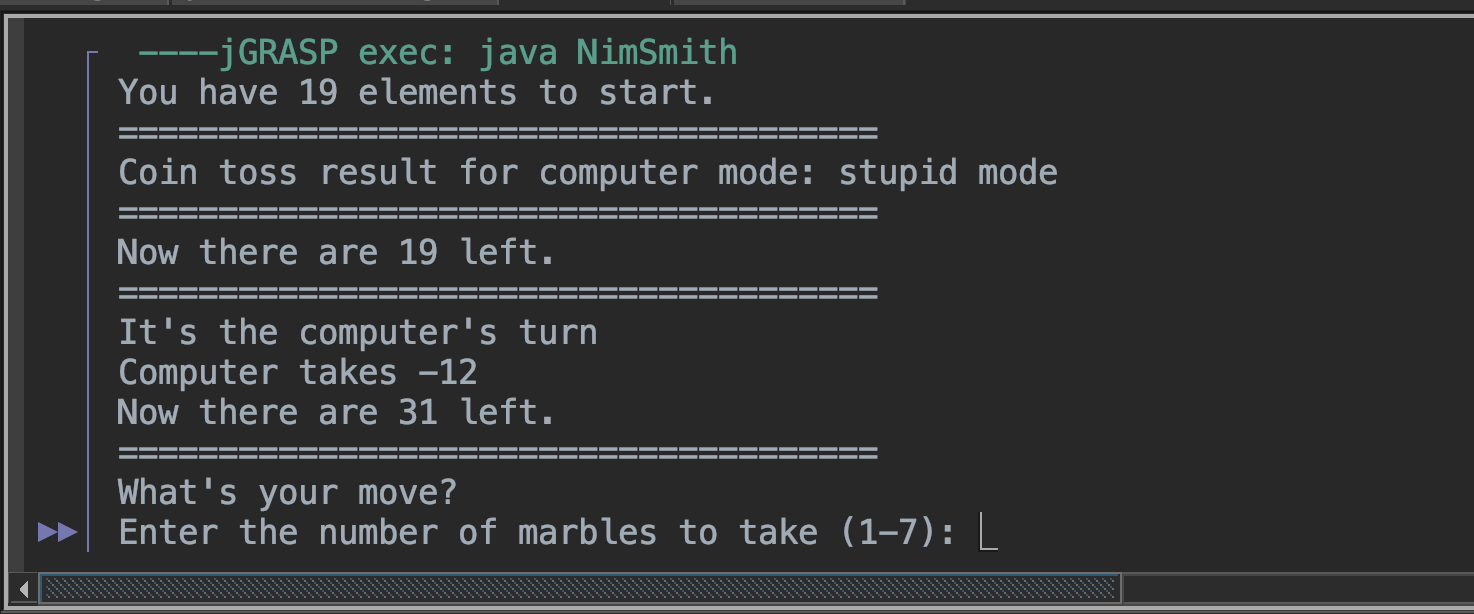
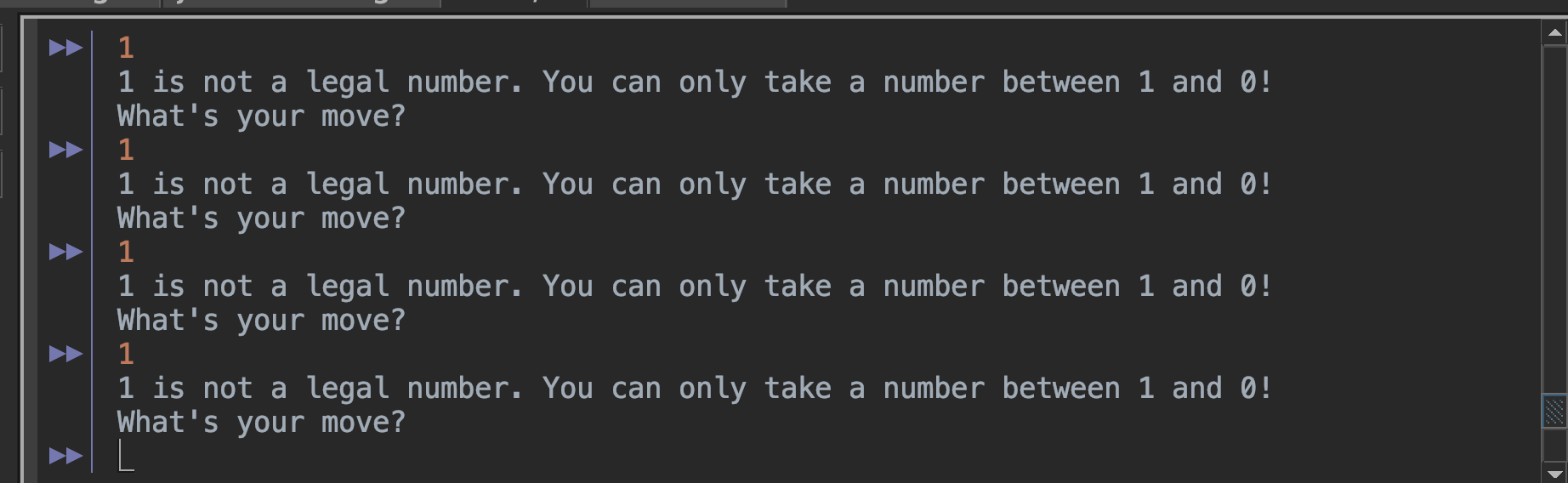
**ERRORS**



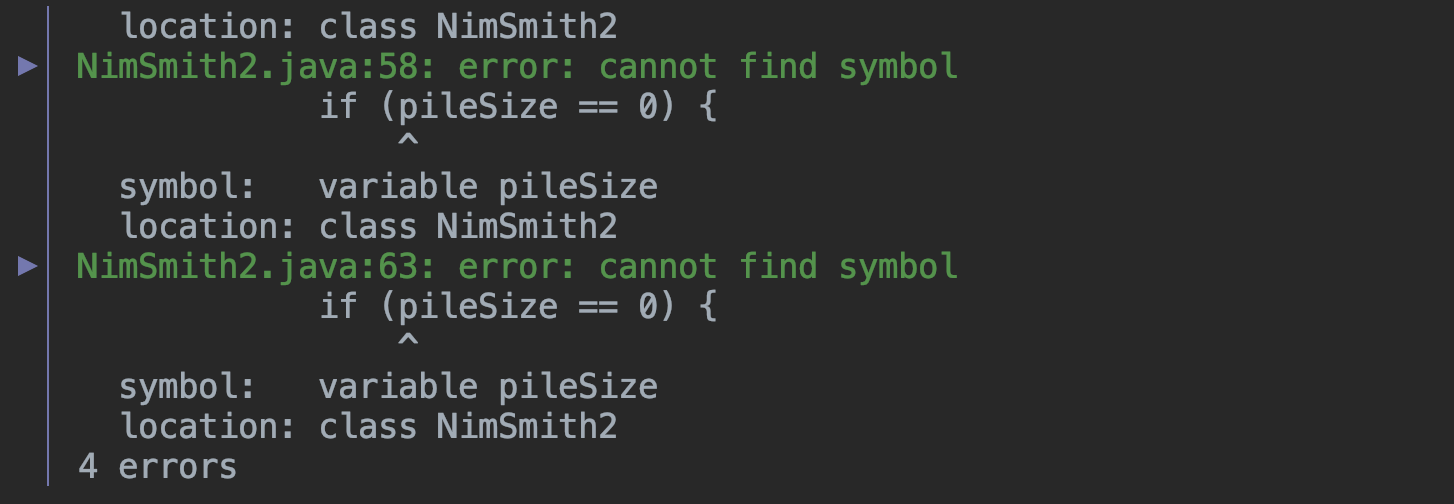
Two methods with same name; just changed one.



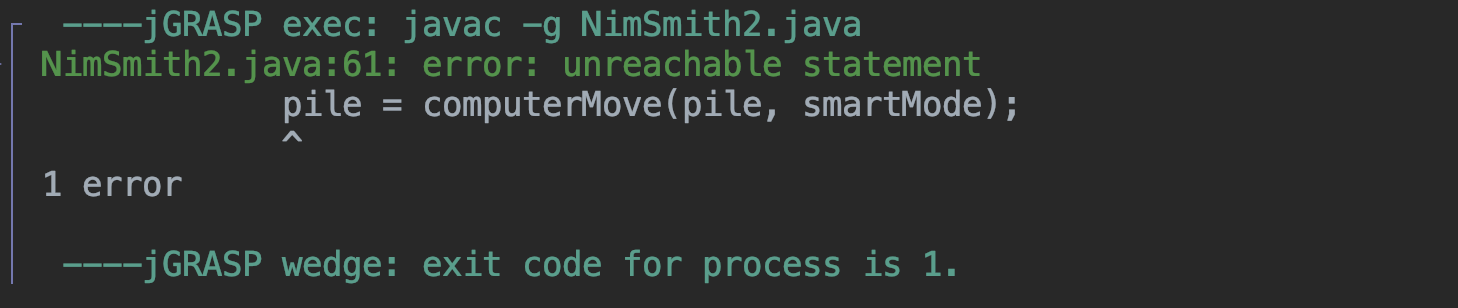
ComputerMove() Method not working properly; issue was with how I was using my Math.Random



Exception with 1 and 0, solved by making a case and including 1.



Variable intializied wrong; Changed to pile



Return statement placed before updating pile; fixed by switching order around

**PSUEDOCODE**

Start

Initialize Scanner scnr

Initialize int pile with a random value between 10 and 100

Initialize boolean computerMove to false

Initialize boolean humanMove to true

Initialize boolean smartMode to false

Initialize boolean stupidMode to true

If a random number between 0 and 1 is 1:

Set computerMove to true

Else:

Set humanMove to true

If a random number between 0 and 1 is 1:

Set smartMode to true

Else:

Set stupidMode to true

While pile is greater than 0:

If computerMove:

Call computerMove(pile, smartMode)

If pile is 1:

Print "Computer won!"

Break

Call humanMove(pile, scnr)

If pile is 1:

Print "You won!"

Break

Else:

Call humanMove(pile, scnr)

If pile is 1:

Print "You won!"

Break

Call computerMove(pile, smartMode)

If pile is 1:

Print "Computer won!"

Break

End

Function computerMove(pile, smartMode):

If smartMode:

Calculate computerMarblesToTake using smartMoveMarbleNum(pile)

Else:

Calculate computerMarblesToTake using stupidMoveMarbleNum(pile)

Subtract computerMarblesToTake from pile

Print "Computer takes [computerMarblesToTake]"

Print "Now there are [pile] left"

Function humanMove(pile, scnr):

Print "What's your move?"

Read marblesToTake from scnr

If marblesToTake is greater than pile / 2 or marblesToTake is less than 1:

Print "[marblesToTake] is not a legal number. You can only take a number from 1 and [pile / 2]!"

Call humanMove(pile, scnr)

Else:

Subtract marblesToTake from pile

Print "Now there are [pile] left"

**TEST RUNS:**

