CS 1632 - DELIVERABLE 5: Performance Testing Conway's Game of Life

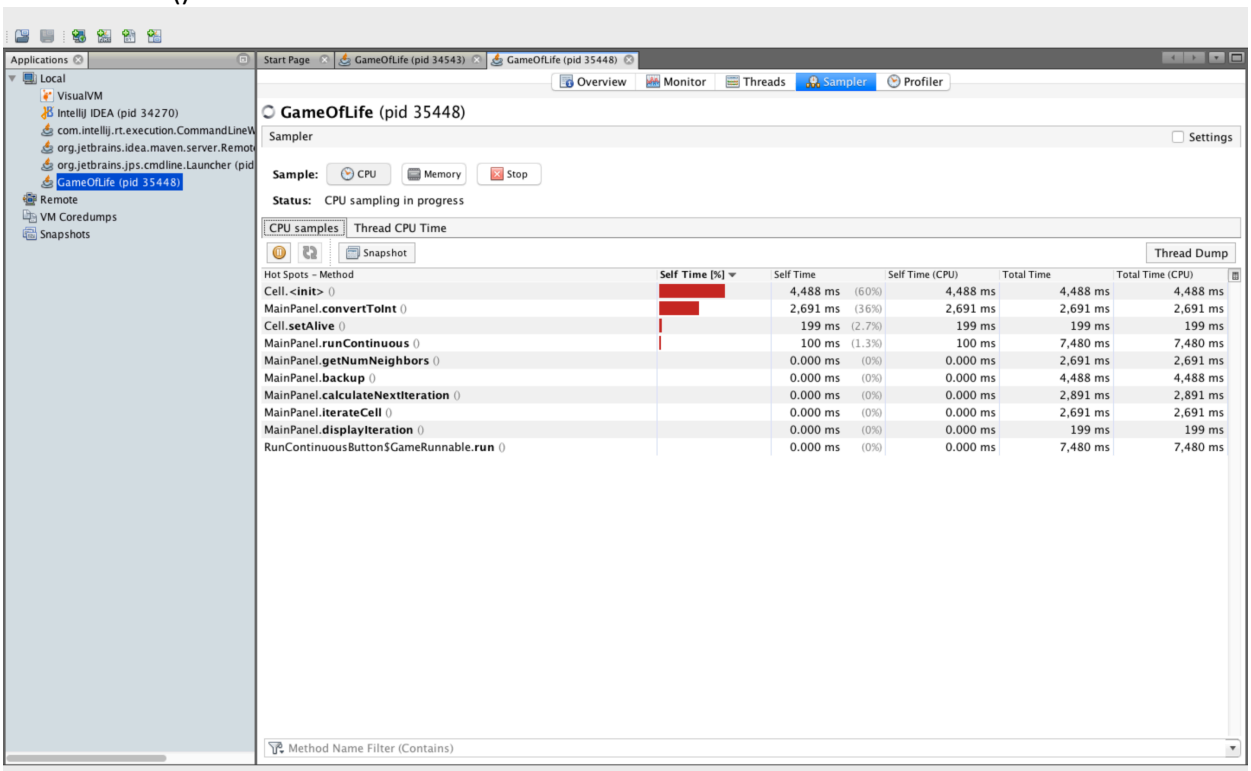
SlowLifeGUI

Evan Sheen

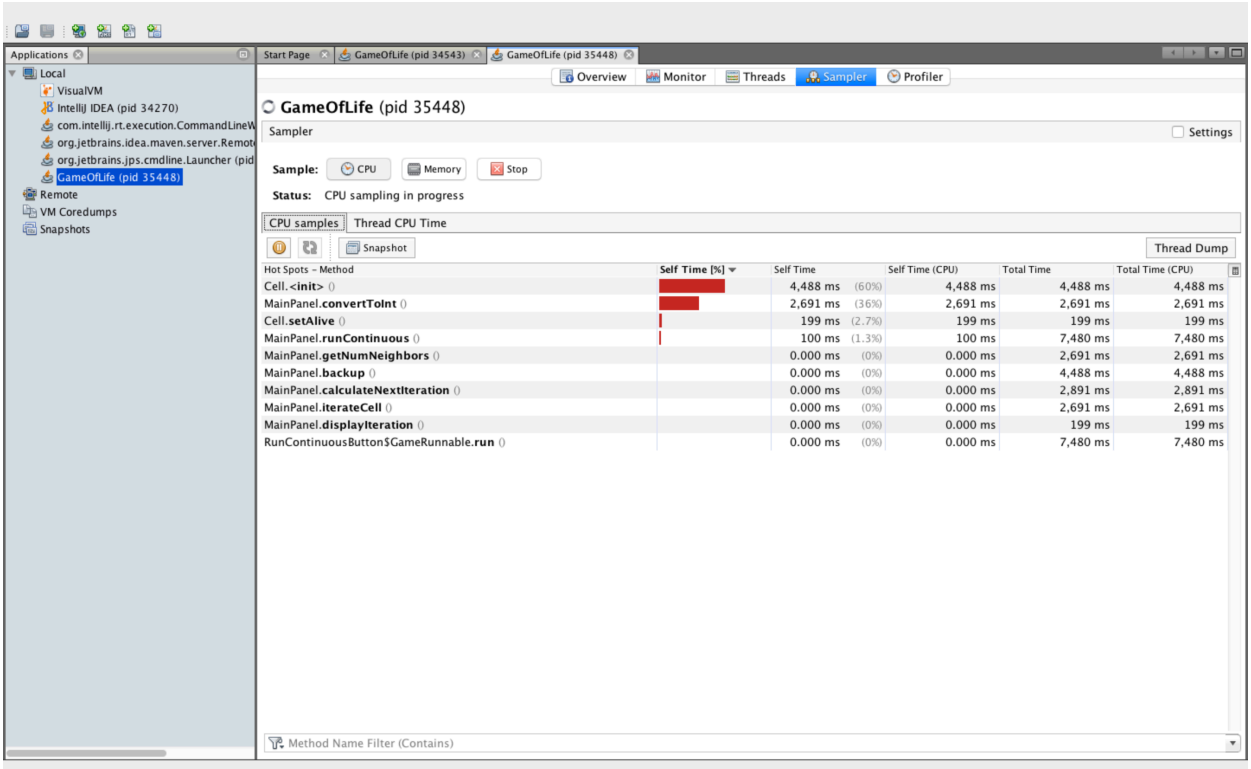
I profiled Conway's game of life with visualVM and the first thing I saw was the convertToInt, runContinous were really eating up a lot of CPU. Inspected the two methods and saw that they could easily be improved. The first one I fixed was convert to int. I removed everything and just returned the passed in integer. For the runContinous method I realized that it was doing anunnecessary for loop. I deleted it. I reprofiled the code after each improvement to make sure it was faster. The last method I choose to fix was the substring method. That simply involved getting rid of the for loop concatenation.

In order to make sure that what I changed was correct I wrote JUNIT tests that asserted that the before method and the after method when passed the same thing produce the same output. The only one I couldn't have a JUNIT automated test for was runContinous() as JUNIT tests for the GUI would be very hard. Manual tests make more sense for testing the GUI because I couldn't figure out how to test GUI objects easily.

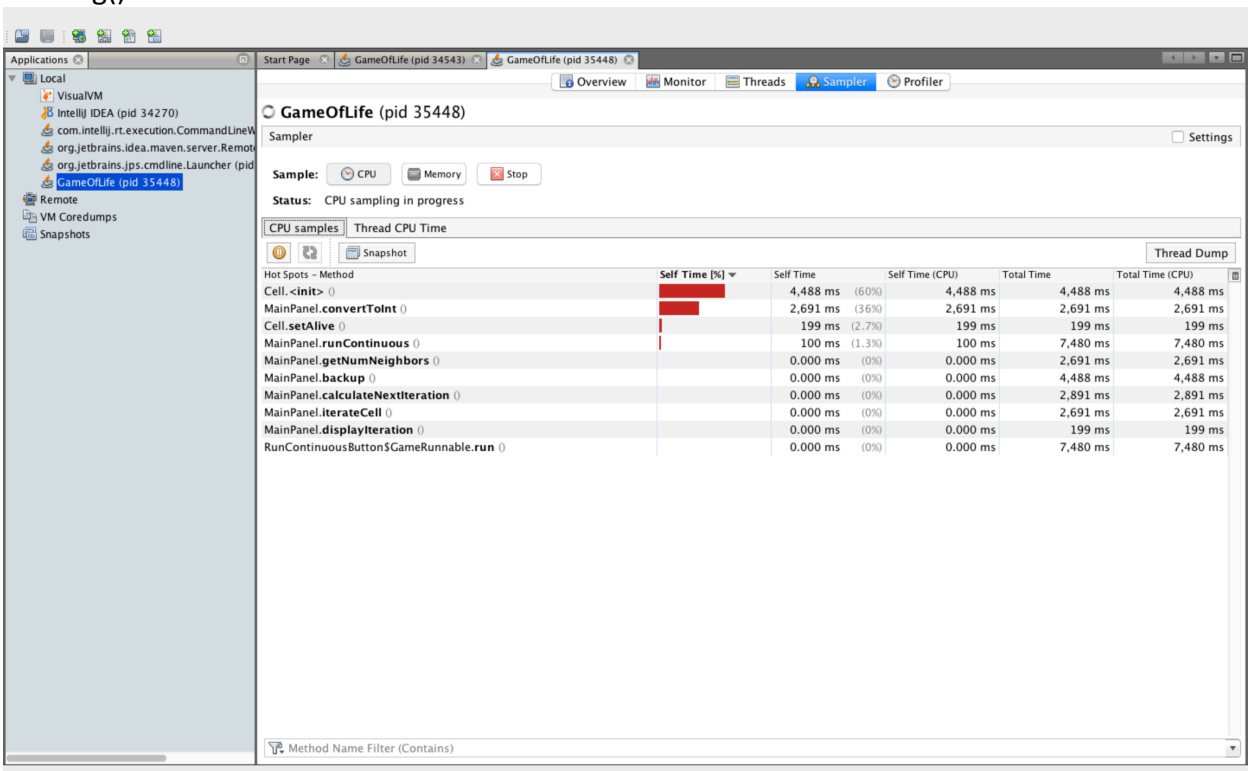
convertToInt() before:



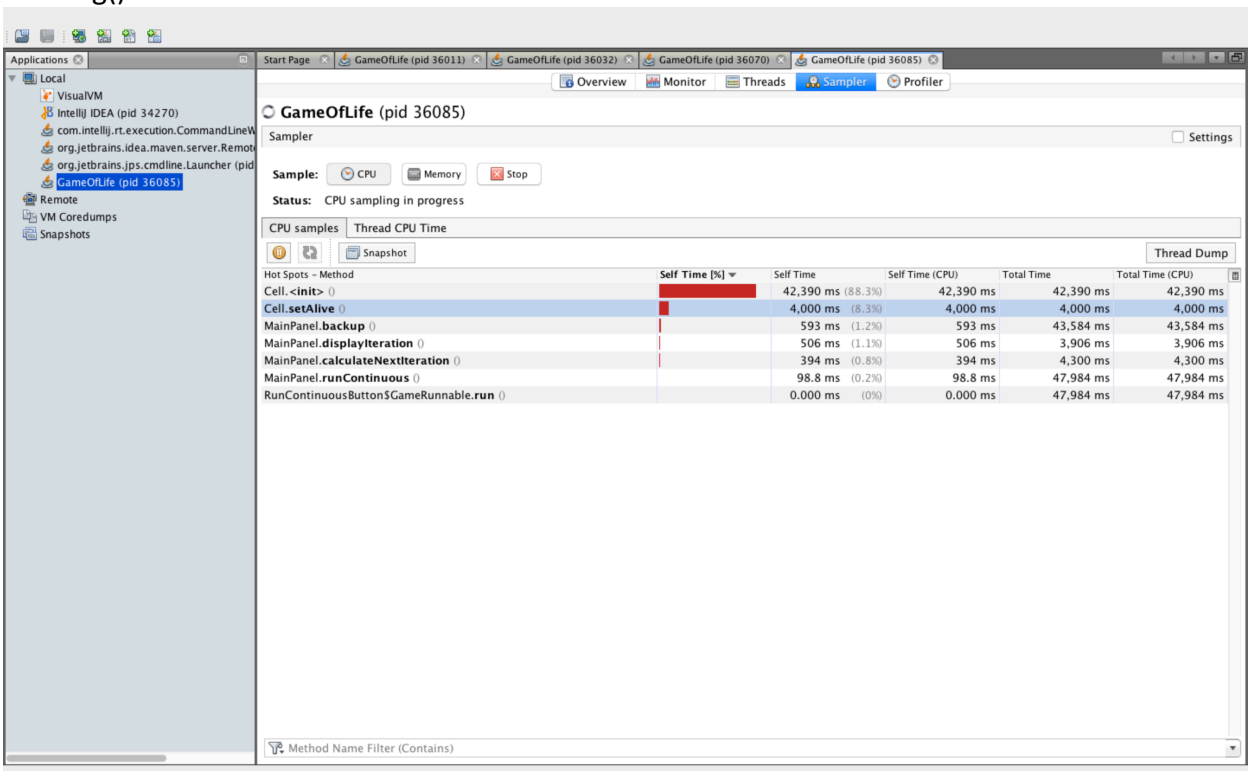
convertToInt() after:



toString() before:



toString() after:



Manual test for runContinous()
Before1:

Before2:

Before3:

After1:

After2:

After3: