JavaLife

Ryan Rahuba (rjr44)

CS 1632 – DELIVERABLE 5: Performance Testing Conway’s Game of Life

**SUMMARY:**

To profile this implementation of Conway’s Game of Life I chose to use VisualVM. To enable an continuous amount of time to evaluate the program I set the World size and iterations to be quite large, 100 and 100000 respectively, so that the program would run for a sufficient amount of time. Once started, and VisualVM was opened to Samples>CPU, it was quite obvious where the hotspot for CPU usage was. According to the profiler’s metrics, World.iterate() was using 77.9% of the CPU.

While reviewing he method in the code base, nothing obvious stuck out. So, I next looked att he method invoked by World.iterate() which included getNumNeighbors(). Going to this method revealed the problem. The if-statements checking to see if a cell’s neighbors were within the array bounds were wrapped in a for-loop with 10,000 iterations. This was extraneous work being done by the CPU as it just performed the same comparisons on the same data that many times. Removing the for-loop showed and instant performance boost with World.iterate () now taking only 1% of the CPU.

Finally I checked runtime for JavaLife to see if the code showed real improvement. To do this I ran the program on a smaller dataset, World size of 20 and only 10,000 iterations. The times returned, shown below, were 83,893 before the change to the code base and 13,517 after the changes.



