



# **CS 1632 Software Quality Assurance**

## **Deliverable 4**

Member 1 Name: Sushruti Bansod

Member 2 Name: Sherryl Augustine

Team Name: S squared

## **1. Introduction**

Write your retrospective here. Please describe the division of work between group members. Also, please describe any difficulties that you faced while using VisualVM.

**Throughout the semester, we have fairly divided work for all the assignments. That was consistent for this deliverable as well. Since Sush had experience running the profiler, she chose to do that and refactor the methods. Due to this, we did not have many difficulties using VisualVM. We then worked together on writing the pinning tests since they were a bit complicated. At one point, we were stuck in figuring out how the unit tests should be written. What helped us the most was being able to ask questions. Here Sherryl was able to understand the concepts better, so she took the lead on completing the remaining work. We were able to finish quickly after those clarifications.**

## 2. Optimized Feature 1

- Write the name of the feature optimized (the label on the button)  
run/run continuous
- Write the name(s) of the method(s) you refactored to optimize this feature  
CalculateNextIteration and IterateCell
- Insert a screenshot of VisualVM CPU profiling **after** optimizing Feature 1. Please include only the “Hot spots” window in the screenshot.

Name	Self Time (CPU)	Total Time (CPU)	Invocations
MainPanel.backup ()	246 ms (47.4%)	274 ms (28.1%)	128
Cell.setAlive (boolean)	90.3 ms (17.4%)	90.3 ms (9.3%)	6,372
Cell.getAlive ()	64.5 ms (12.4%)	64.5 ms (6.6%)	31,773
MainPanel.iterateCell (int, int)	57.4 ms (11%)	151 ms (15.6%)	3,175
MainPanel.getNumNeighbors (int, int)	31.6 ms (6.1%)	96.1 ms (9.8%)	3,175
MainPanel.calculateNextIteration ()	29.4 ms (5.7%)	240 ms (24.6%)	127
MainPanel.displayIteration (boolean[][])	0.0 ms (0%)	59.0 ms (6%)	127

### 3. Optimized Feature 2

- a. Write the name of the feature optimized (the label on the button)

Write

- b. Write the name(s) of the method(s) you refactored to optimize this feature

Cell.toString()

- c. Insert a screenshot of VisualVM CPU profiling *after* optimizing Feature 2. Please show only the profile for Feature 2 (do not invoke Feature 1 while profiling). Please include only the “Hot spots” window in the screenshot, as before.

Name	Self Time (CPU)			Total Time (CPU)		Invocation
Cell.toString ()		0.0 ms	(-%)	0.0 ms	(-%)	
Cell.getAlive ()		0.0 ms	(-%)	0.0 ms	(-%)	
FileAccess.saveFile (String, String)		0.0 ms	(-%)	0.0 ms	(-%)	
WriteButton\$WriteButtonListener.actionPerformed (java.awt.event.ActionEvent)		0.0 ms	(-%)	0.0 ms	(-%)	
MainPanel.toString ()		0.0 ms	(-%)	0.0 ms	(-%)	