Performance-based Learning Outcomes
CSSE373 – Formal Methods

1. Demonstrate a thorough understanding of string theory
2. Predict the results of an operation based on its contract
3. Write JUnit-type test cases for a component's operations based on model and operation's specs
4. Implement an operation using design-by-contract where the calling operation is responsible for satisfying the requires clause
5. Implement an operation based solely on its specification and the specs of the component(s) it uses
6. Implement a layered component based on the component's model, external and internal contracts (abstraction function and representation invariant), and based on the layered upon component's model and external contracts
7. Fill in a reasoning table for an operation (no branching) utilizing the operation's requires/ensures clause and the requires/ensures clauses for all operations called by the operation
8. Write down VCs based on filled-in reasoning table
9. Prove VCs generated from a filled-in reasoning table
10. Show that defective operation's proof cannot go through given its VCs and a filled-in reasoning table
11. Prove a loop invariant
12. Prove a recursive operation
13. Given a loop invariant for a correctly implemented loop, identify which part of the loop invariant
14. Explain differences between a traditional compiler and a verifying compiler
15. Explain differences between a traditional programming language and a programming language that supports mechanical verification