Samuel Bailey

Week 2 Journal

CS-320

* What is static testing?

Static testing is the first step in debugging code. Static testing is when the programmer/tester reviews the code without running the code. The tester/programmer just does a readthrough of the code to understand and readthrough what is going on. Usually this is where we ask questions like, “is this what the user is looking for?”. Also adding more comments in the testing phase. We also review overall design here like data structures and algorithms are reviewed here to find the best option.

* What is dynamic testing?

This is the testing that we all think of when we think of the testing phase, or at least I did. This phase the code is actually ran and we try different tests to make the code throw weird errors based on input. As a rule this is one of the most important phases because if a program is opened to the public it will be abused. No if, ands, or buts. This testing is called Dynamic based on the dynamics of the change of user input.

* What are the differences between static and dynamic testing?

These testing phases/models are completely different in the best ways. In static we are reviewing the layout and creation of the code to see if it is the best option. Dynamic testing is running a sort of hypothesis of what the code will do. In science we imagine testing as running a series of actions to see if what we think will happen is correct, that’s similar to dynamic testing.

* Why is it important to use both static and dynamic testing?

All programmers/developers should always check both static and dynamic testing, both of these phases test completely different and go over different parts of the program. For example, someone can skip the static tests and still have the program work, however there could be some written bugs that still allow the program to work but allow for hackers to get in, or leaked information, or worse an inefficient program.