Cpt S 422

TERM: FALL 2016

INSTRUCTOR: EVAN OLDS

EMAIL: EVANOLDS@WSU.EDU

Websites

- Blackboard
  - http://learn.wsu.edu
- EECS Page
  - http://eecs.wsu.edu/~eolds

This is not a "Software Testing" Class

- From the course catalog:
  - "Dependable software systems; software verification and validation, testing; CASE environments; software management and evolution."
- Testing will be one of many themes, but this is still software engineering
- A strong technical understanding of what happens when code executes is vital
  - Without understanding how a system works, you won't fully understand all the ways a system can break
  - Without understanding how a system can break, you won't fully understand how to test that system

Course Topics

- Software correctness/accuracy
- Scientific calculations and number formats
- Finding and fixing bugs
- Testing terminology and concepts
- Design processes
- Planning, specification, maintenance and documentation
- Designing flexible software that can be easily extended for automated testing
- Testing project

Lecture Notes

- In class only
- Attendance required – if you miss a class you must meet up with another student (or group of students) from the class and discuss the lecture content from the day(s) you missed
- From the WSU academic regulations:
  - **UNIVERSITY SPONSORED**. Any student who is required to participate in off-campus, university-sponsored activities such as field trips, musical performances, judging teams, intercollegiate athletic events, etc., should obtain an official Class Absence Request form from the faculty or staff member supervising the off-campus activity
  - (there's a somewhat similar thing for members of the military)

Lecture Format

- I will lecture on topics, but I don't want it to be just me talking for 50 minutes straight each lecture. The desire is to make it as interactive as possible.
- Students are expected to be asking questions when they have them and offering responses to questions when they are asked.

Prerequisite Knowledge

- C#
  - 321 (from fall 2016 on)
  - 322 (prior to fall 2016)
- Linux
- Some topics we cover will be generic and extend across multiple programming languages, others will be specific to a particular programming language
- All coding assignments will be in C# and will be tested/graded on a Linux platform

Prerequisite Knowledge

- Object-oriented programming
- Inheritance and polymorphism
- Time and space complexity analysis
- Data structures from 122 and 223
- (basically knowledge from all the core prerequisites)

Homework Assignments

- Two places to look:
  - EECS Page
  - Blackboard
- Many (all?) coding assignments will require you to make a class library (DLL)
  - Zip ONLY code (.cs) files in a zip with no nested folders
  - Submit zip online to EECS page
  - The only thing on Blackboard will be tests/quizzes. Do not submit coding assignments here.
- Remaining details are on syllabus