

Test Driven Development (TDD)

Best Practices Workshop, March 25-26 2019, Hampton VA

Carlos Cruz
Jules Kouatchou
Brent Smith

NASA GSFC Code 606/610 (ASTG/GMAO)
Greenbelt, Maryland 20771

What is TDD?

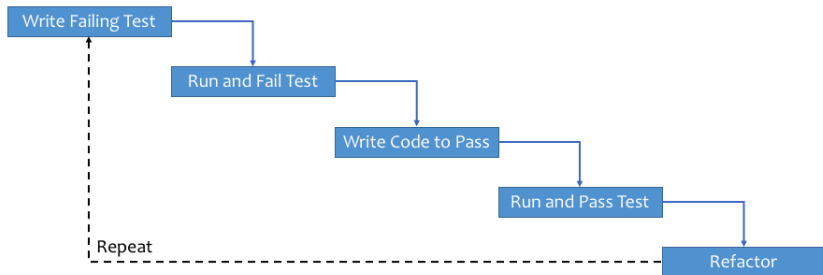
An **agile** methodology (a style of programming) where coding, testing, and design are tightly interwoven and in which unit **T**ests **D**rive the **D**evelopment of the production code.



How does TDD work?

Have requirements handy.

Break up your task into functional units, then for each unit:



Demo



FAQ

What makes a good unit test?

A good test should verify that one logical path through a method works.



FAQ

How many tests should I have?

The minimum amount that lets you write all the production code. The minimum amount, because every test slows down refactoring (when you change production code, you have to fix all the failing tests). On the other hand, refactoring is much simpler and safer on code under tests.



FAQ

How should I order my tests?

DON'T !!!!!

Every unit test should be completely independent of every other unit test. You'll have to build-up any necessary objects and context to make that specific unit testable.



FAQ

Should test coverage be 100%?

No. Sometimes you can avoid using TDD on some parts of a project.



FAQ

I am able to write code with very a few bugs,
I don't need testing.

You may able to to that, but is the same consideration valid for all your team members? They will eventually modify your code and break it. It would be nice if you wrote tests so that a bug can be spotted immediately and not in production.



The End

Questions?

