Test Driven Development

What?

Test-driven development (**TDD**) is a software development process that relies on the repetition of a very short development cycle: requirements are turned into very specific test cases, then the code is improved so that the tests pass. This is opposed to software development that allows code to be added that is not proven to meet requirements

Why?

InspecaoRegular/IrItemEstruturaInspecao/ObterInformacoesAuxiliaresDaAnomalia

Why?

```
if (requisicao.IdDirecaoLeituraInstrumento == 0)
    v_UltimoNivelSegurancaInferior = nivelSegurancaInferiorRepositorio.ObterUltimoNivelSegurancaInferiorDoInstrumentoPorDirecao(requisicao.IdInstrumento, requisicao.IdDirecaoLeituraInstrumento);
else
    v_UltimoNivelSegurancaInferior = nivelSegurancaInferiorRepositorio.ObterUltimoNivelSegurancaInferiorInstrumento(requisicao.IdInstrumento);
// Esse cara precisa de férias
```


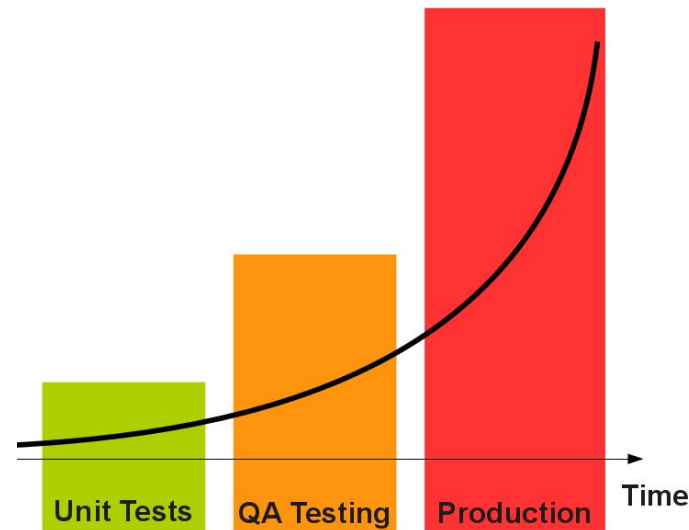
System development trend: a bad outcome

Growth of code lines count based on a product version
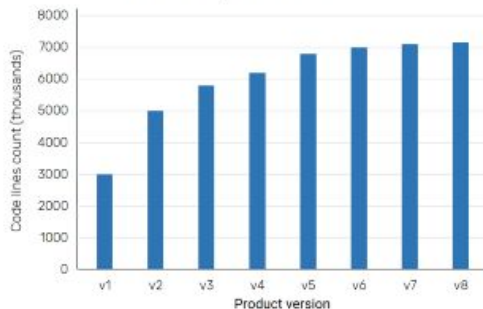
Increase in the cost of a line of code based on a product version
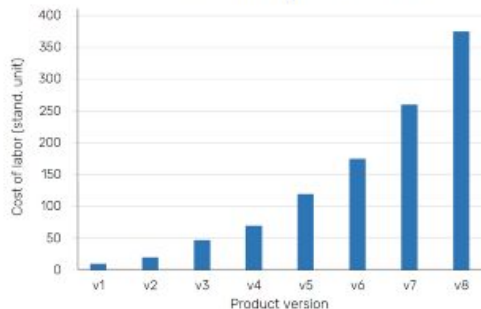
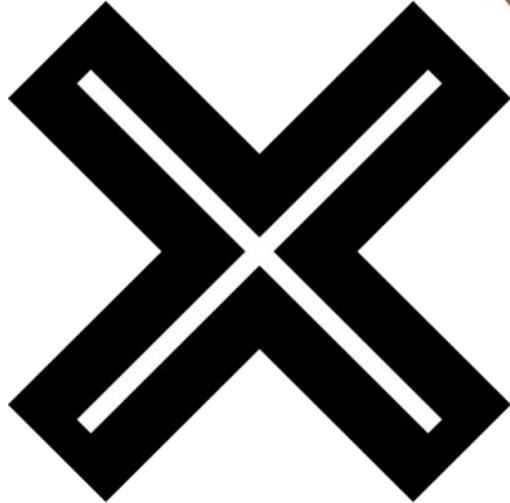**Conclusion:** in a while, product development will be too expensive because of its complexity. We should stay away from this type of situation.

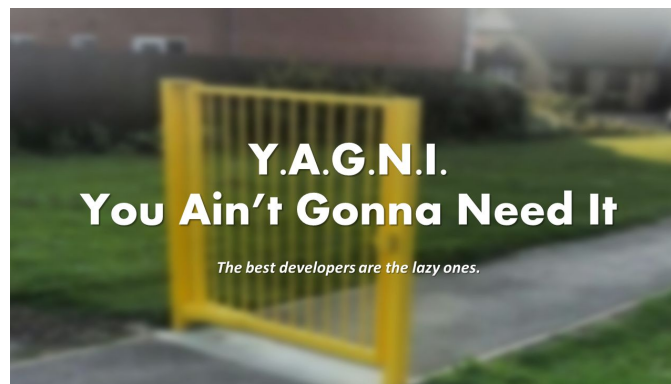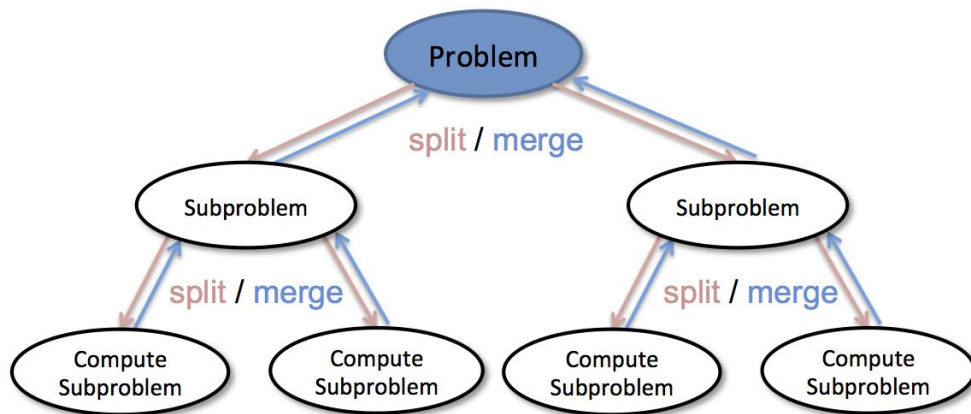Unit Tests · QA Testing · Production · Time

How?

Manual Tests

The Three Laws of TDD

1. You are not allowed to write any production code unless it is to make a failing unit test pass.
2. You are not allowed to write any more of a unit test than is sufficient to fail, and compilation failures are failures.
3. You are not allowed to write any more production code than is sufficient to pass the one failing unit test.

Y.A.G.N.I.
You Ain't Gonna Need It

The best developers are the lazy ones.

TDD Cycle: TEST FAILS → TEST PASSES → REFACTOR