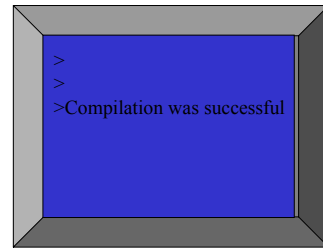
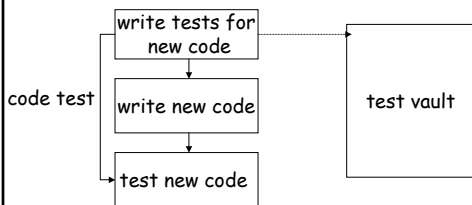


Test Plans

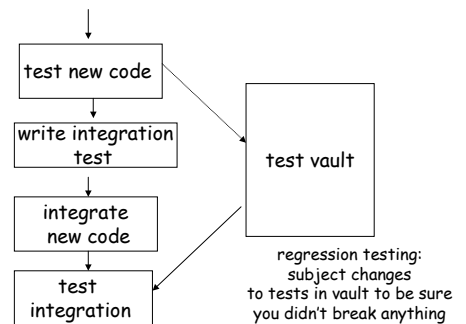


This does not mean the code is debugged!

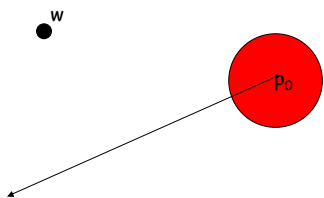
Test Driven Development I



Test Driven Development II



Unit Test for Vertex Collision



Test Cases

- You know answer ahead of time
- Tests include problem cases

Example: Step 1

w



p₀

choose w and p₀ randomly

Are there special cases we should consider or ignore?

Example Step 1

w



p₀

choose w and p₀ randomly such that

1. distance < r — more on this later
2. distance = r
3. distance > r

Example Step 2

w



p₀

randomly set the ball's velocity

are there special cases to consider or ignore?

Example Step 2

w



r- ϵ

r

r+ ϵ

p₀

randomly set the ball's velocity so that it just hits and just misses

Questions

- Are these good test?
- Are there other kinds of tests we should do?
- How can we automate testing?
 - Test generation
 - Running the tests
- How will we do regression testing?

Numerical inaccuracies

the ball's position at time of collision may have real but not rational coordinates. what impact does this have?

w

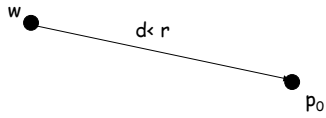


r

p₀

randomly set the ball's velocity so that it just hits and just misses

When the impossible happens



what should you do?

KNOW WHAT TO EXPECT!!

Come up with a test plan for Prototype II.
How will you evolve the plan over the course of the project?
Implement the plan.