In Berkley lecture 12, we see the following screen:

Unit- and Functional-Level Characterization Tests
* Cheat: write tests to learn as you go
  -- See Screencast 8.3.1 at screencast.saasbook.info

it "should calculate sales tax" do
  order = mock('order')
  order.compute_tax.should == -99.99
end

# object 'order' received unexpected message 'get_total'

it "should calculate sales tax" do
  order = mock('order', :get_total=> 100.00)
  order.compute_tax.should == -99.99
end

# expected compute_tax to be -99.99 was 8.45

it "should calculate sales tax" do
  order = mock('order', :get_total=> 100.00)
  order.compute_tax.should == 8.45
end

A) What is the above showing us?
B) Why is this called a 'cheat'?
C) What testing system is running the above tests?

[see other question on back of this page]
What are some examples of important software metrics (and what do you know about what they are measuring)? How do software metrics figure into handling legacy code according to the Berkeley lectures we have seen?