

4/15/6

11/19/15

continuing w/ Patton ch. 7
white box

if time after dealing w/ midterm =

data coverage

data includes variables, constants,
arrays, data structures,
database tables, key board
& mouse inputs, files,
screen output, devices I/O,
network I/O

data flow - track data as
it moves through sw system

can be done at unit,
integration or full system level

"taint" tracking used for sensitive
data, particularly for mobile
- does sensitive data
even ~~app~~ go from, say,
address book to
network via unexpected app

11/19/15

use debugger to check intermediate values of data ul in method (black box only, see at begin & end)

try to get "interesting" values

for example, consider code that computes compound interest for some financial application

$$A = P(1 + r/n)^{nt}$$

P - principal r - annual interest rate
 n - # times per year interest compounded
 t - # years A - amount after time t

a value $n=0$ will cause equation to blow up because of divide by zero

black box testing would try input $n=0$ - but what if n is not an input or method parameter, but derived from other values. what does test ~~for~~ to force $n=0$ situation?

4156

11/19/15

error forcing

- can use debugger to force variables to specific values that might be hard to achieve from regular testing

but make sure not to create situation that can never happen, e.g.
setting n to zero AFTER
the code that checks whether
 n is zero (invalid test case)

main goal of error forcing is to try to
force all error messages & also
try to find cases when there
should be error message but isn't

let's say your sw is designed to
handle 2,048 network connections
- do you set up 2,049 connections?

4156

11/19/15

code coverage

discussed
previously

test program states &
control flow among them

enter & exit every method,
execute every line of code (statement)
follow every logic & decision
path (branch)

code-coverage analyzer tools
runs while test cases are running
(often by instrumented
the target code &/or
the test code)

- each time method,
statements, branch
executes, this is recorded
- enable statistics on what
portions were executed
& what not
- need to write additional
tests to force coverage

4156

11/19/17

branch coverage \neq condition coverage

make each conditional
expression T vs. F
independently

- ① if x AND y
- ② do thrs;
- ③ do that;

test 1 - x & y both true
achieves full statement
coverage, but we're not done

test 2 - x true, y false

test 3 - x false, y true

test 4 - x & y both false

} any of
these
would
cover
other
side
of
branch