

## ***Today's Agenda***

- **Black Box Testing**

## ***Planning a Black Box Test Case***

- **Look at requirements/problem statement to generate.**
- **Said another way: test cases should be traceable to requirements.**
  
- **ID of test case**
- **Describe/enumerate test input conditions**
- **Expected results**
- **Actual Results**
  
- **When should you do this?**
- **What about automation?**

## ***Important Consideration for Black Box Test Planning***

- The inputs must be very specific
- The expected results must be very specific
- If the test case fails, you don't want to have to work \*too\* hard to re-run the test when the code is (supposedly) fixed
- **You must write the test case so anyone on the team can run the exact test case and get the exact same result/sequence of events**
- **Example: "Students who receive a grade of 70 or higher pass the exam."**

## ***What Kind of Triangle?***

- Given 3 triangle side, classify it's type: equilateral, isosceles, scalene.
- Precondition: The 3 sides do form a triangle.

<i>ID</i>	<i>Description/ Input</i>	<i>Expected Results</i>	<i>Actual Results</i>

## Monopoly

- Go to OS site
- Download Monopoly game into an Eclipse project
  - See [“Getting started with Eclipse” tutorial](#)
- Run in test mode (see Monopoly Test Plan) and execute some test cases.
- Comments?

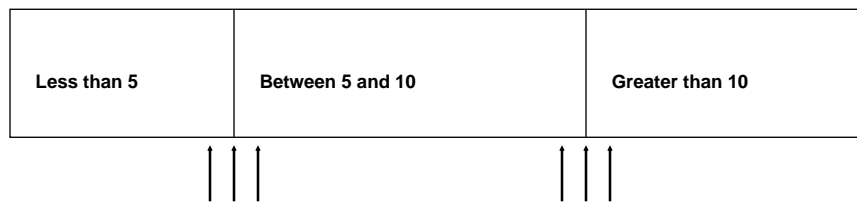
## Equivalence Class Partitioning

- Divide your input conditions into groups (classes).
- Input in the same class should behave similarly in the program.
- **Be sure to test a mid-range value from each class.**
- Example: “Integer between 5-10, inclusive”
- Test input values like: 3, 7, and 14

Less than 5	Between 5 and 10	Greater than 10
-------------	------------------	-----------------

## Boundary Value Analysis

- Focus on boundaries . . . Because a greater number of errors tend to occur at the boundaries of the input domain
  - Range input, a to b, test with a, b, a-1, a+1, b-1, b+1 (if integer range, otherwise, slightly less than a and slightly more than b)
  - If you can only have a certain quantity (q) of something, try to create q-1, q, q+1



NC STATE UNIVERSITY

## Decision Table Testing

	Rule 1	Rule 2	Rule 3
<b>Conditions</b>			
A	Yes	Yes	No
B	Yes	No	--
<b>Actions</b>			
C	Yes	No	Yes

Each column becomes a test case.

NC STATE UNIVERSITY

## Monopoly Decision Table

- *If a Player (A) lands on property owned by another player (B), A must pay rent to B. If A does not have enough money to pay B, A is out of the game.*

	Rule 1	Rule 2	Rule 3
<b>Conditions</b>			
A lands on B's property	Yes	Yes	No
A has enough money to pay rent	Yes	No	--
<b>Actions</b>			
A stays in game	Yes	No	Yes

## Dirty/Failure Test Cases

- **Can something cause division by zero?**
- **What if the input type is wrong (You're expecting an integer, they input a float. You're expecting a character, you get an integer.)?**
- **What if the customer takes an illogical path through your functionality?**
- **What if mandatory fields are not entered?**
- **What if the program is aborted abruptly or input or output devices are unplugged?**

**Think diabolically!**

## Donald Knuth



*My test programs are intended to break the system, to push it to its extreme limits, to pile complication on complication, in ways that the system programmer never consciously anticipated. To prepare such test data, I get into the meanest, nastiest frame of mind that I can manage, and I write the cruelest code I can think of; then I turn around and embed that in even nastier constructions that are almost obscene.*

## Monopoly II

- Look at each test case. Mark with E for equivalence class, B for Boundary Value Analysis, D for diabolical.
- For each requirement challenge yourself to make the test cases more complete. Write the detailed test cases.

## ***Your Turn . . . Write some test cases (using the template)***

- **A teacher wants to compute an average of up to 10 test scores.**
  
- **Students must input their first name, last name, and social security number. Their first and last names must be no more than 15 characters each. Standard SSN format NNN-NN-NNNN.**