### **Boundary Value Testing**





- Input domain testing is the most commonly taught (and perhaps the most commonly used) software testing technique
- There are a number of approaches to boundary value analysis
- We will study some of the limitations of domain testing



#### **Boundary Value Analysis**

- Many programs can be viewed as a function F that maps values from a set A (its domain) to values in another set B (its range)
- The input variables of F will have some (possibly unstated) boundaries:

$$F: A \to B$$

$$a \le x_1 \le b \qquad c \le x_2 \le d$$



#### Boundary value analysis – 1

- What is boundary analysis?
- What is the rationale for boundary analysis?



#### Boundary value analysis – 2

- For each variable, select five values
  - Min The minimum
  - Min+ Slightly above the minimum
  - Non Nominal
  - Max— Slightly below the maximum
  - Max Maximum



#### Critical assumption

- What is the critical assumption made with boundary value testing?
- Based on this assumption
  - How are test cases selected?



#### Single fault assumption

- Failures are only rarely the result of the simultaneous occurrence of two (or more) faults
- Generate test cases as such for all i
  - Values of all but one variable x<sub>i</sub> at nominal
  - x<sub>i</sub> assumes all 5 values from the previous slide
    - Figure 5.2 in textbook for two variable case
- What are the number of test cases?



#### Two-variable function test cases

$$< x_{1nom}, x_{2min} > < x_{1min}, x_{2nom} > < x_{1nom}, x_{2nom} > < x_{2nom}$$

#### Apply BVA to the Triangle problem

$$1 \le a \le 200$$
  
 $1 \le b \le 200$   
 $1 \le c \le 200$ 

# Advantages

When does boundary value analysis work well?



#### Advantages – 2

- Independent variables
  - Single fault assumption
- Physical quantities
- Languages that are not strongly typed
  - Why were strongly typed languages developed?

## Limitations

What are the limitations of boundary value analysis?

# Limitations – 2

- Does not work well for Boolean variables
  - Why are these not suitable?
- Does not work well for logical variables
  - PIN, transaction type
  - Why are these not suitable?
- When variables are not independent i.e. are dependent
  - What example does the textbook give?
- Not that useful for strongly-typed languages



#### Variations of boundary value analysis

- What extensions or variations are made for boundary value analysis?
- What is the justification for each?



- Robustness testing
- Worst case testing
- Robust worst case testing
- Special value testing
- Random testing



#### Robustness testing

- Add two more values per variable
  - Max+ Slightly greater than the maximum
  - Min— Slightly less than the minimum
- What is the expected output?
  - Hopefully error message, system recovers
- Implementing these test cases may not be possible
  - What is the difficulty?
- What are the number of test cases?
- When is robust testing mandated?



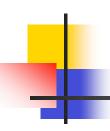
#### **Worst-Case Testing**

- Rejects the simple fault assumption and tests all combinations of values
- Often leads to a large number of test cases with low bug-finding power
  - Why?
- Usually better to apply Special Value Testing
  - test cases based on the tester's intuition
- What are the number of test cases?



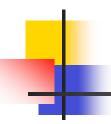
#### Robust worst case testing

- Add the values min— and max+ to the possible variable values
- Now take all combinations of variable values
- What are the number of test cases?



#### Special value testing

- Use best engineering judgment
  - Intuition
  - domain knowledge
  - Experience
  - Soft spots



### In class activity

Do exercises 1, 2 and 3



- Select random values for each variable
- How many tests do we make?