Classification Tree Method

by
Ramraj S ME.,
Assistant Professor
Department of Software Engineering

2 - Mar -2017

Agenda

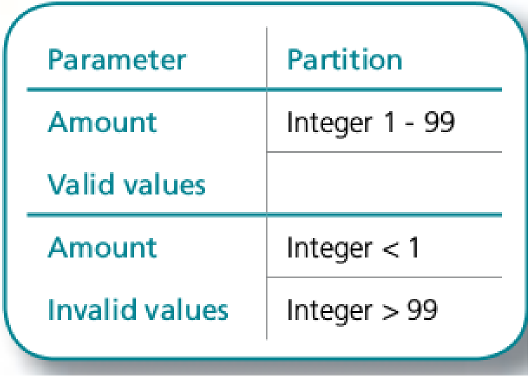
- Classification Tree Method
- Boundary Value analysis
- Classification of values
- CTM - Tree with Boundary value analysis

Classification Tree Method

- The objective of the CTM is to find a minimal, non-redundant (but sufficient) set of test cases by trying to cover several aspects in a single test case.
- A tree with more leaf classes naturally results in more test cases than a tree with less leaf classes.
- The number of leaf classes needed at least for a given tree is called the "minimum criterion".
- each leaf class should be marked (mapped) in at least one test case.
- "maximum criterion" can be calculated, which gives the maximal number of test cases for a given classification tree

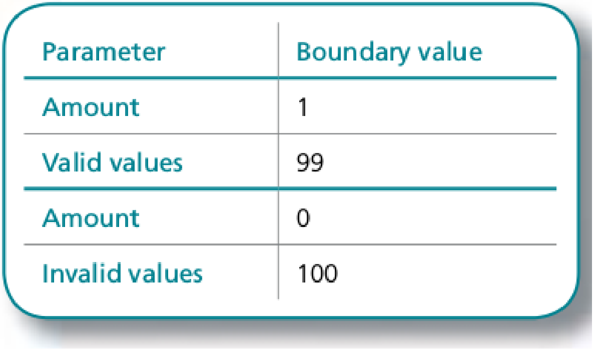
Boundary Value analysis

- Boundary Value Analysis determines the effectiveness of test cases for a given scenario
- Concentrates on the Corner cases or the boundaries of the input domain rather than its center
- For example If we want to test a field from 1 to 9 boundary values will be 0,1,2,and 8,9,10
- The left hand value is called the lower boundary ie 1 here in this example and the Right side value 9 is called the Upper boundary
- The idea behind boundary values analysis is contrary to equivalence partitioning, because one method takes a set of values as equivalent and the other method prefers special values in such a set.
- Both approaches can be expressed in the CTM



Parameter	Partition
Amount	Integer 1 - 99
Valid values	
Amount	Integer < 1
Invalid values	Integer > 99

Figure : Before Boundary Value Analysis



Parameter	Boundary value
Amount	1
Valid values	99
Amount	0
Invalid values	100

Figure : After Boundary Value Analysis

-5678 ... -1, 0, 1 ... 17 ...

Identifying test relevant aspects

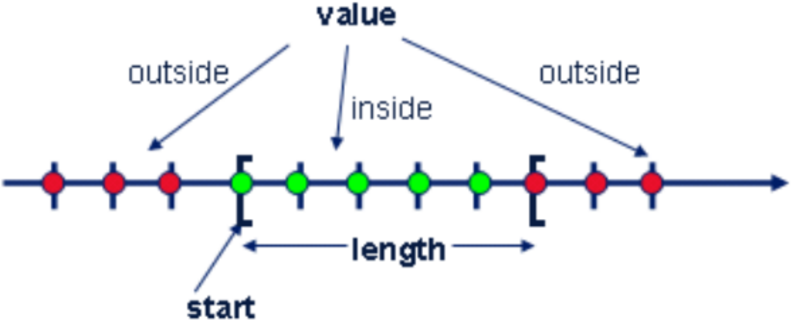


Figure : Is-value-Inrange

Atleast a test case should be there for testing each of these aspect

- Range of Values
  - Start
  - Length

Forming classes

- classes are formed for the base classifications according to the equivalence partitioning method.
- E.g. if the problem specification would state "if the start value is greater than 20, the length value doubles", we should form a class for start values greater than 20 and a class for start values smaller than resp. equal to 20.
- for our problem specification, classes will be formed like this:

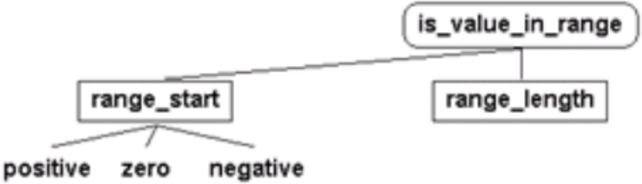


Figure : Is-value-Inrange further developed

Range specification

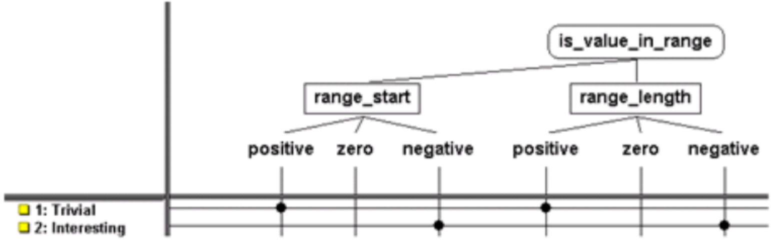


Figure : Is-value-Inrange further developed

- One marker selects the class "positive" forthe start of the range; the other marker selects the class "positive" for the length of the range.
- range with the start value of, say, 5 and a length of, say, 2 would be according to this specification
- such a value pair most probably would have been used in any case when test cases for the problem at hand are thought of. Therefore, this first specification was named "Trivial"

References I

[1] Frank Buechner *"TestCaseDesignUsingTheClassificationTreeMethod "*

Thank you