Software Quality Assurance and Testing (SQAT)

White Box Testing (Part 2)

dr. Joost Schalken-Pinkster
Windesheim University of Applied Science
The Netherlands

joost@schalken.me

The contents of these course slides is (in great part) based on:
Neil Taylor (2019) *Course presentations for Software Quality Assurance and Testing (SQAT)*, Aberystwyth University.

Introduction

Continuing to think about White Box Testing:

- Control Flow
- Cyclomatic Complexity
- Basis Paths

Learning Objectives

- Discuss the use of cyclomatic complexity and basis paths to guide the process to select test cases.

Control Flow Graphing & Basis Paths

- Control Flow Graphs can be used to understand the connections within the code
  - How one part of the code moves to the next part
  - Helps us to understand the control structure
  - Cyclomatic Complexity to measure function complexity
    - Help for testing

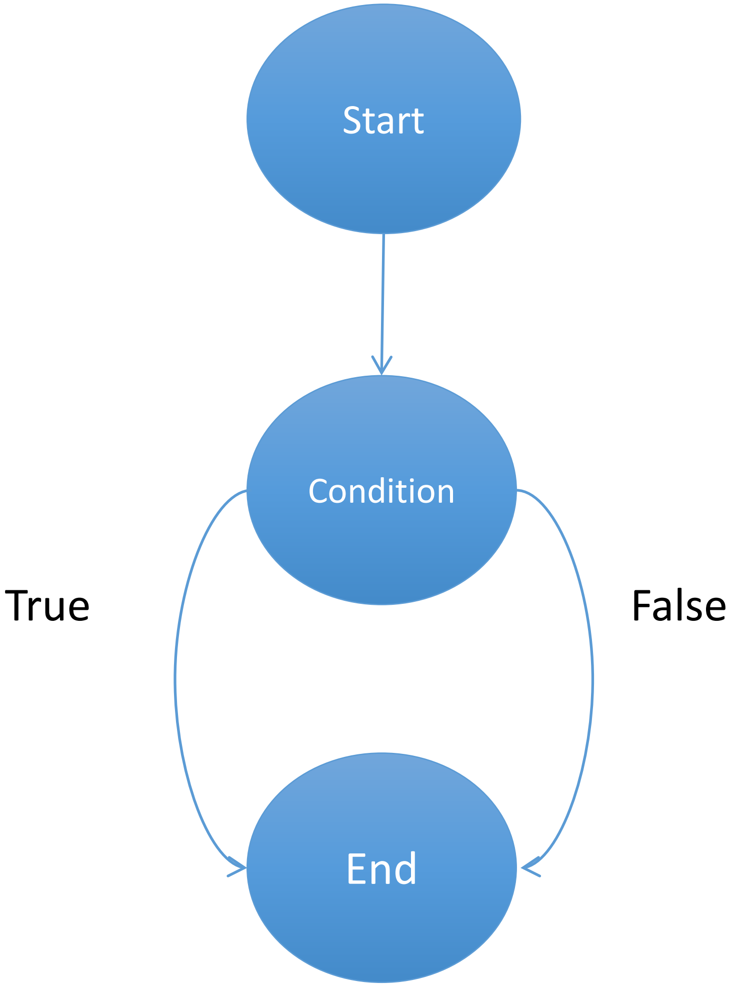
- Basis Path is an independent path through
  - Each path represents a path to test
  - Developed by T McCabe & A H Watson

Control flow graph

Contains:

- Nodes – represents actions in the code. We are particularly interested in conditions that change the flow in the code.
- Edges – Arrows that link the Nodes.
- Start and Entry Point

```
public String
getSubstring(String s, int p)
{
    String result = null;
    if(s.length > p) {
        result = s.substring(p);
    }
    return result;
}
```



Cyclomatic Complexity

- Calculation that gives an indication of complexity
- Higher the number, the more complex the code is
- Two similar ways to calculate
  - Number of Edges – Number of Nodes + 2
  - Number of condition clauses + 1
- Can use the calculation for testing to indicate a minimum number of tests for a section of code

```
public String getSubstring(String s, int p)
{
    String result = null;
    if(s.length > p) {
        result = s.substring(p);
    }
    return result;
}
```

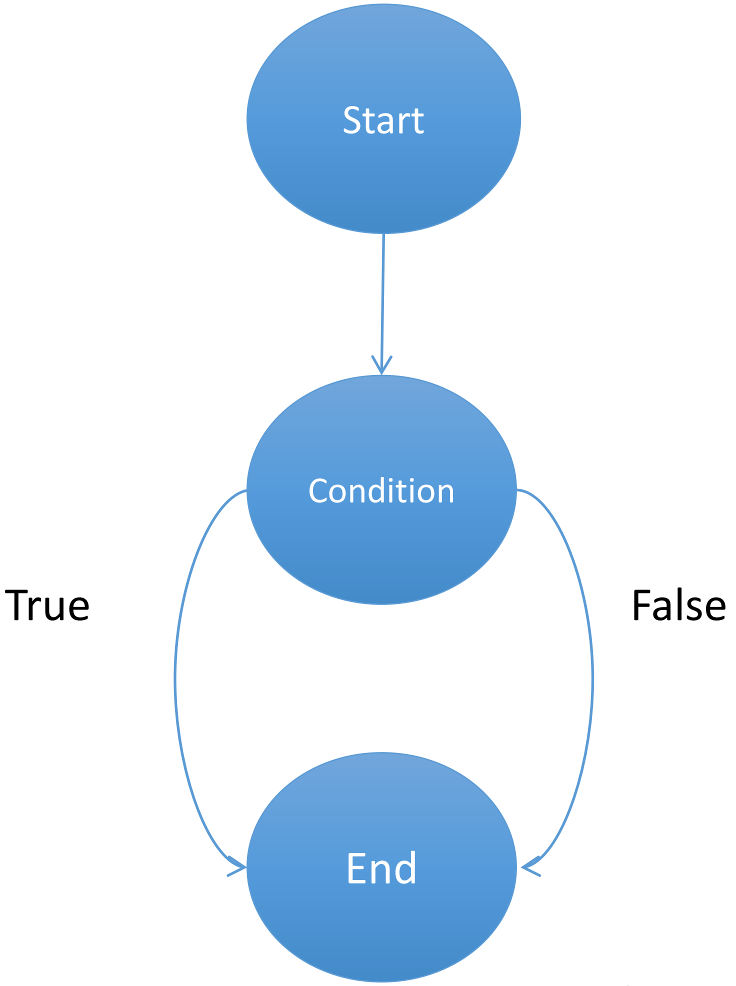
Cyclomatic Complexity:

Either:

$$E - N + 2 \rightarrow 3 - 3 + 2 = 2$$

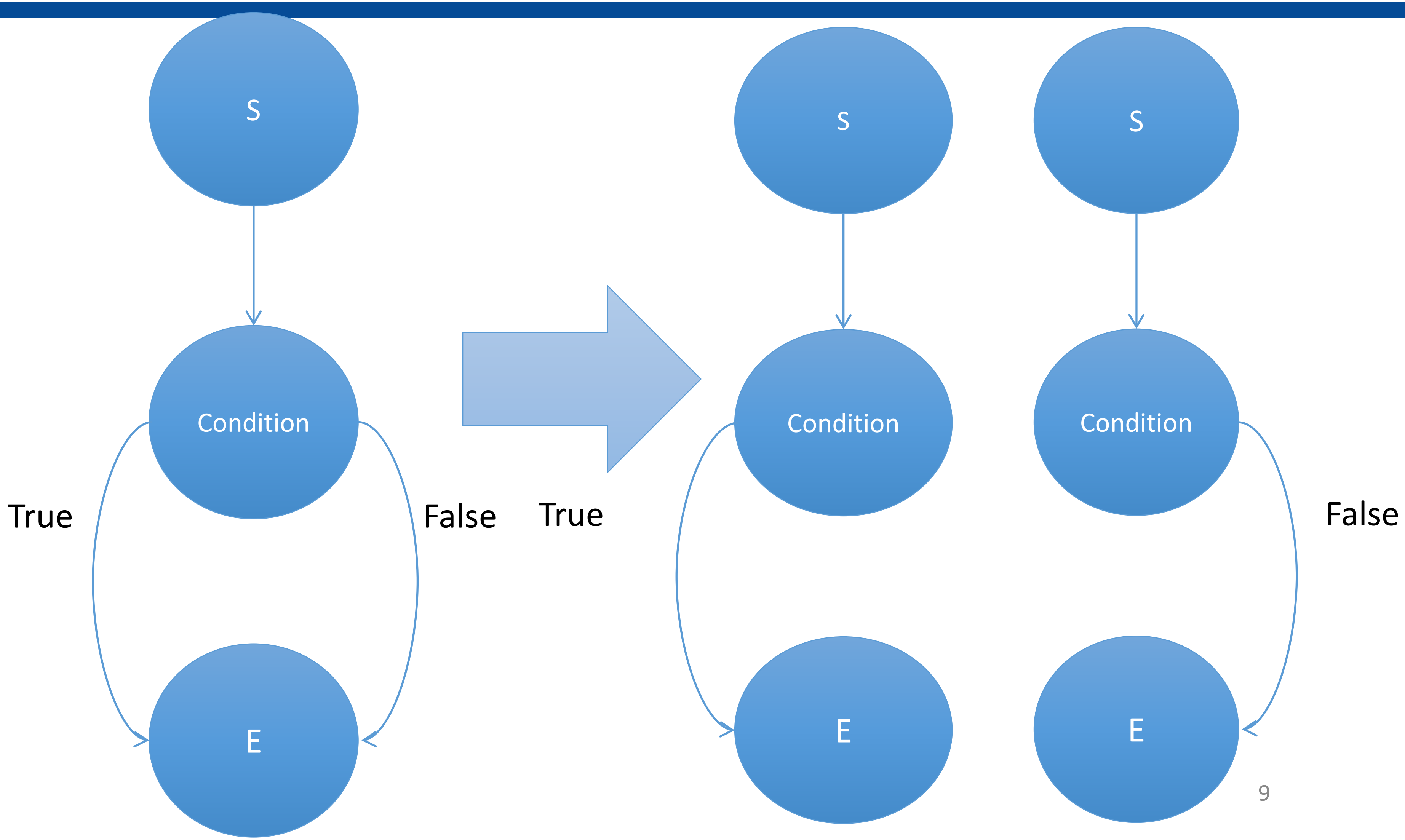
Or

$$C + 1 \rightarrow 1 + 1 = 2$$



Basis Path

- Independent path through the code
- We should test each independent path



Example

Example taken from “*How We Test Software At Microsoft*”

- A function counts the instances of the letter C in strings that begin with a letter A

```
    private static int CountC(string myString){
A0     int index = 0, i = 0, j = 0, k = 0;
       char[] strArray = myString.ToCharArray();

A1     if(strArray[index] == ‘A’) {
A2        while(++index < strArray.Length) {
A3           if(strArray[index] == ‘B’) {
                j = j + 1;
             }
A4           else if(strArray[index] == ‘C’) {
                i = i + j;
                k = k + 1;
                j = 0;
             }
          }
          i = i + j;
       }
A5     return i;
    }
```

Decision Table

Two test cases can be used to evaluate each conditional clause to True or False at least once.

ID	Input	A1	A2	A3	A4	Expected	Actual
1	D	F	-	-	-	0	0
2	ABCD	T	T & F	T & F	T & F	1	1

Test Case 2 follows the path:

A0 → A1(T) → A2(T) → A3(T) → A2(T) → A3(F) → A4(T) → A2(T) → A3(F) → A4(F) → A2(F) → A5

Do these test cases exercise all of the paths?

Basis Paths

- Cyclomatic Complexity for this section of code?
  - Diagram drawn on board
- What is the calculation?

Basic Path Table for CountC

Basis Path	Path	Input	Expected
1	A0 → A1(F) → A5	D	0
2	A0 → A1(T) → A2(F) → A5	A	0
3	A0 → A1(T) → A2(T) → A3(T) → A2(F) → A5	AB	0
4	A0 → A1(T) → A2(T) → A3(F) → A4(T) → A2(F) → A5	AC	1
5	A0 → A1(T) → A2(T) → A3(F) → A4(F) → A2(F) → A5	AD	0

Truth Table for CountC

Test	Param	A1	A2	A3	A4	Expected	Actual
1	D	F	-	-	-	0	0
2	A	T	F	-	-	0	0
3	AB	T	T & F	T	-	0	1
4	AC	T	T & F	F	T	1	0
5	AD	T	T & F	F	F	0	0

This set of tests highlight some problems in the code. We don’t get the correct values for test cases 3 and 4

Summary

- Looked at a way to think about the complexity of a piece of code
- Used that to guide the number of test cases that we should think about
- Considered an example from the Microsoft book, which illustrates the use of the technique.

Any Questions?