

SENG 275

SOFTWARE TESTING

DR. NAVNEET KAUR POPLI

DEPT. OF ELECTRICAL AND COMPUTER
ENGINEERING



PATH COVERAGE



Path coverage

- With branch+condition coverage, we looked at each condition and branch individually. Such a criterion gives testers more branches to generate tests, especially when compared to the first criterion we discussed (line coverage).
- However, although we are testing each condition to be evaluated as true and false, this does not ensure testing of all the paths that a program can have.
- Path coverage does not consider the conditions individually. Rather, it considers the **(full) combination of the conditions in a decision**. Each of these combinations is a path. You might see a path as a unique way to traverse the CFG.



Calculation

- The calculation is the same as the other coverages:

$$\text{path coverage} = \frac{\text{paths covered}}{\text{paths total}} \cdot 100$$



See the following example that focus on a small piece of the count method:

```
if (!Character.isLetter(str.charAt(i)) & (last == 's' | last == 'r'))  
{  
    words++;  
}
```

The decision in this if-statement contains **three** conditions and can be generalized to $(A \ \& \ (B \ | \ C))$, with:

$A = \text{!Character.isLetter(str.charAt(i))}$

$B = \text{last} == \text{'s'}$

$C = \text{last} == \text{'r'}$

To get 100% path coverage, we would have to test all the possible combinations of these three conditions.



University
of Victoria

We make a truth table to find the combinations:

Tests	A	B	C	Outcome
1	T	T	T	T
2	T	T	F	T
3	T	F	T	T
4	T	F	F	F
5	F	T	T	F
6	F	T	F	F
7	F	F	T	F
8	F	F	F	F



Full path coverage may be expensive

- This means that, for full path coverage, we would need 8 tests just to cover this if statement. It is a large number for just a single statement.
- Imagine programs that have multiple conditions or rely on loops.
- By aiming at achieving 100% path coverage, testers can indeed come up with good tests.
- However, the main issue is that achieving 100% path coverage might **not** always be **feasible** or it might be **too costly**.
- The number of tests needed for full path coverage will grow **exponentially** with the number of conditions in a decision.



