Prime Path Origins

Jeff Offutt

Topics

- Test goals
- Why criteria?
- Number of paths in a graph
- Dealing with loops
- Prime path coverage
- My favorite coverage criteria
- How to be a professional

Test goals or purpose

Each individual test must have a purpose

- What goal is it trying to achieve?
- What kind of mistake is it trying to find?
- What fact is it trying to prove or disprove?

The goal must be documented

Coverage criteria

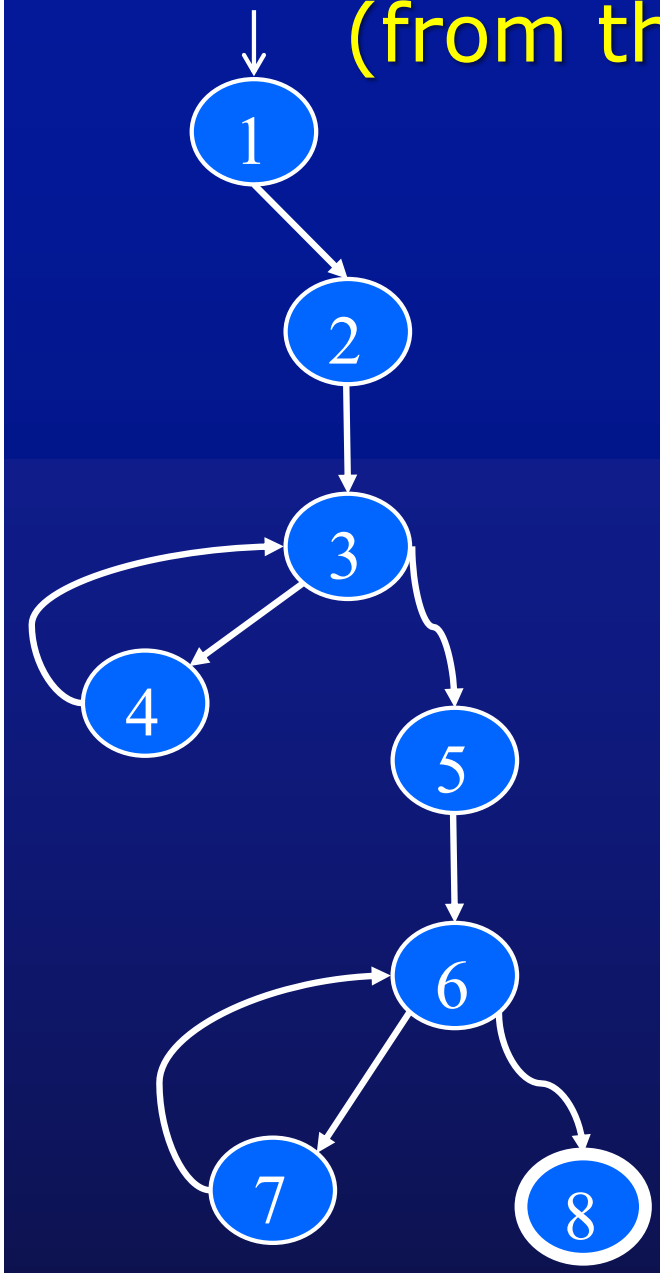
- Even small programs have too many inputs to fully test them all
  - private static double computeAverage (int A, int B, int C)
  - A, B, and C each has more than 4 billion possible values
  - Over 80 octillion possible tests!!
  - Input space might is effectively infinite
- Testers search a huge input space
  - Trying to find the fewest inputs that will find the most problems
- Coverage criteria give structured, practical ways to search the input space
  - Search the input space thoroughly
  - Not much overlap in the tests

Coverage criteria on graphs

- Node coverage (eg, CFGs, call graphs)
  - Statement coverage, method coverage, component coverage, …
- Edge coverage
  - Branch coverage, call coverage, …
- Edge-pair coverage
  - Every adjacent pair of edges (or 3-node subpaths)
  - Often forces unusual cases like empty lists, null objects, items not found in a data structure, …
- Prime paths
  - All unique paths in a graph
  - Cover loops without having an infinite number of paths

How many paths?

(from the method *computeStats*())



Paths in this graph:

[1, 2, 3, 5, 6, 8]

[1, 2, 3, 4, 3, 5, 6, 8]

[1, 2, 3, 5, 6, 7, 6, 8]

[1, 2, 3, 4, 3, 5, 6, 7, 6, 8]

But we have 2 loops …

[3, 4, 3]

[6, 7, 6]

How many times do we go around the loops?

Dealing with loops

- 1970s : Execute cycles once ([3, 4, 3] in previous example), informal
- 1980s : Execute each loop, exactly once (formalized)
- 1990s : Execute loops 0 times, once, more than once (informal description)
- 2000s : Prime paths (touring, sidetrips, and detours)

Prime path coverage requires loops to be :

- skipped
- executed once, and
- executed more than once

If you satisfy prime paths, you also satisfy node coverage, edge coverage, and edge-pair coverage

Why I love prime path coverage

- PPC is an elegant way to get a very high amount of coverage
- The definitions are a little complicated, but it's easy to automate
  - It relies on deep graph theory, but you don't need to be good at graph theory to use PPC
- Paul Ammann and I invented PPC because we were frustrated with the old ways to handle loops

My favorite coverage criteria (and why)

1. Edge-pair coverage
   - Very simple to understand and visualize
   - Only a few more tests than branch coverage
   - The additional tests explore some very subtle interactions where problematic bugs tend to hide
2. Prime path coverage
   - Elegant mathematical way to handle loops
   - Yields a very strong set of tests
   - Widely used in financial industry
3. Bypass testing
   - Very simple idea, but very powerful
   - Useful for testing and for security
   - Built into Selenium

Suggestions for professionalism

- Show integrity
  - Be honest, not nice
  - If you can't do it, don't promise it (under-promise & over-deliver)
  - Tell the truth, not what you think they want to hear
- Learn how to work in teams
  - Respect your teammates
  - Know your strengths, your weaknesses, and their strengths
  - Disagree without being disagreeable
- Respect competence, not confidence

Good team members

… are reliable

… can be trusted

… criticize in private and praise in public

… are flexible

… put the team first

… accept and live the vision of the group

… cooperate instead of compete

Habits of confident people

- They ask questions instead of judging
  - "*Be curious, not judgmental*" – Walt Whitman
- They're not afraid to be wrong—and admit it freely
- They listen more than they talk
  - Bragging is a sign of insecurity
- They give credit to others—they don't need validation
- They ask for help without hesitation
  - A huge part of confidence is knowing your limits
- They avoid criticizing others
- They own their mistakes—using them to learn and teach
- They seek out people who are better and smarter than they are

8 phrases heard in healthy teams

- I trust you
- Thank you!
- You've got this
- I'm here for you
- I made a mistake
- That's okay, we can fix it
- Your ideas are valuable
- What's your perspective?

Summary

You are lucky to have the opportunity to join the strongest industry

If you don't know how to test it, don't build it

Believe the data, not your wishes