

Activity: Prime Path Coverage

**CS 3250
Software Testing**

Prime Paths and Simple Paths

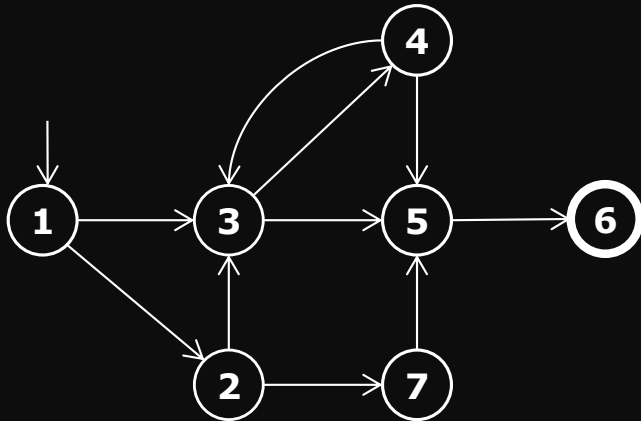
Prime path

- Simple path that is **not subpath** of any other simple path

Simple path

- Path that does not have an internal loop

List Test Requirements for Prime Paths



! Indicate a final node or no more extending

* Indicate cycle (or loop), no more extending

Start with path of length = 0

[1]
[2]
[3]
[4]
[5]
[6]!
[7]

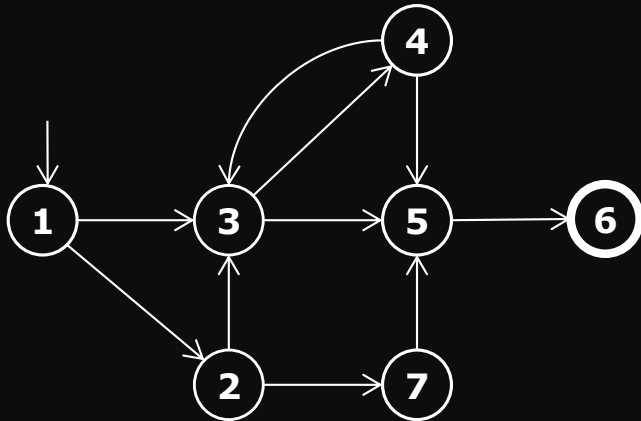
Extend path to length = 1

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

Extend path to length = 2

[1, 2, 3]
[1, 2, 7]
[1, 3, 4]
[1, 3, 5]
[2, 3, 4]
[2, 3, 5]
[2, 7, 5]
[3, 4, 3]*
[3, 4, 5]
[3, 5, 6]!
[4, 3, 4]*
[4, 3, 5]
[4, 5, 6]!
[7, 5, 6]!

List Test Requirements for Prime Paths



! Indicate a final node or no more extending

* Indicate cycle (or loop), no more extending

Extend path to length = 2

[1, 2, 3]
 [1, 2, 7]
 [1, 3, 4]
 [1, 3, 5]
 [2, 3, 4]
 [2, 3, 5]
 [2, 7, 5]
 [3, 4, 3]*
 [3, 4, 5]
 [3, 5, 6]!
 [4, 3, 4]*
 [4, 3, 5]
 [4, 5, 6]!
 [7, 5, 6]!

(from previous page)

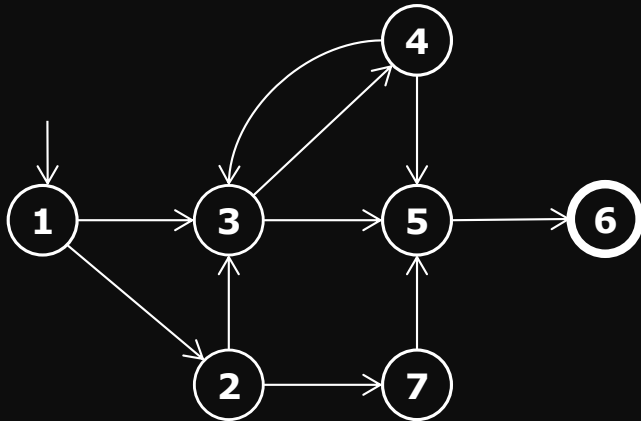
Extend path to length = 3

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

Do not include [1, 3, 4, 3] since it is not a simple path and thus is not a prime path

Do not include [2, 3, 4, 3] since it is not a simple path and thus is not a prime path

List Test Requirements for Prime Paths



! Indicate a final node or no more extending

* Indicate cycle (or loop), no more extending

Extend path to length = 3

[1, 2, 3, 4]

[1, 2, 3, 5]

[1, 2, 7, 5]

[1, 3, 4, 5]

[1, 3, 5, 6]!

[2, 3, 4, 5]

[2, 3, 5, 6]!

[2, 7, 5, 6]!

[3, 4, 5, 6]!

[4, 3, 5, 6]!

Extend path to length = 4

[1, 2, 3, 4, 5]

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

[1, 2, 7, 5, 6]!

[1, 3, 4, 5, 6]!

[2, 3, 4, 5, 6]!

Extend path to length = 5

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

(from previous page)

Determine Prime Paths

Consider paths with ! and * (ignore the remaining paths)
Start with the longest path, cross out paths that are its subpaths
Paths in are prime paths

[1, 2]	[1, 2, 3]	[1, 2, 3, 4]	[1, 2, 3, 4, 5]	[1, 2, 3, 4, 5, 6]!
[1, 3]	[1, 2, 7]	[1, 2, 3, 5]	[1, 2, 3, 5, 6]!	
[2, 3]	[1, 3, 4]	[1, 2, 7, 5]	[1, 2, 7, 5, 6]!	
[2, 7]	[1, 3, 5]	[1, 3, 4, 5]	[1, 3, 4, 5, 6]!	
[3, 4]	[2, 3, 4]	[1, 3, 5, 6]!	[2, 3, 4, 5, 6]!	
[3, 5]	[2, 3, 5]	[2, 3, 4, 5]		
[4, 3]	[2, 7, 5]	[2, 3, 5, 6]!		
[4, 5]	[3, 4, 3]*	[2, 7, 5, 6]!		
[5, 6]!	[3, 4, 5]	[3, 4, 5, 6]!		
[7, 5]	[3, 5, 6]!	[4, 3, 5, 6]!		
	[4, 3, 4]*			
	[4, 3, 5]			
	[4, 5, 6]!			
	[7, 5, 6]!			

Identify Test Paths

For each prime path, check if it starts with an initial node and ends at a final node.
If not, extend the path to start with initial nodes and end at final nodes

Already start
with initial nodes
and end at final
nodes.
They can already
be used as test
paths

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

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

[1, 2, 7, 5, 6]!

[1, 3, 4, 5, 6]!

[1, 3, 5, 6]!

[4, 3, 5, 6]!

[3, 4, 3]*

[4, 3, 4]*

Test paths

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

[1, 2, 3, 5, 6]

[1, 2, 7, 5, 6]

[1, 3, 4, 5, 6]

[1, 3, 5, 6]

[**1**, **3**, 4, 3, 5, 6]

[**1**, **3**, 4, 3, 4, **5**, **6**]

Some test paths can tour multiple test requirements