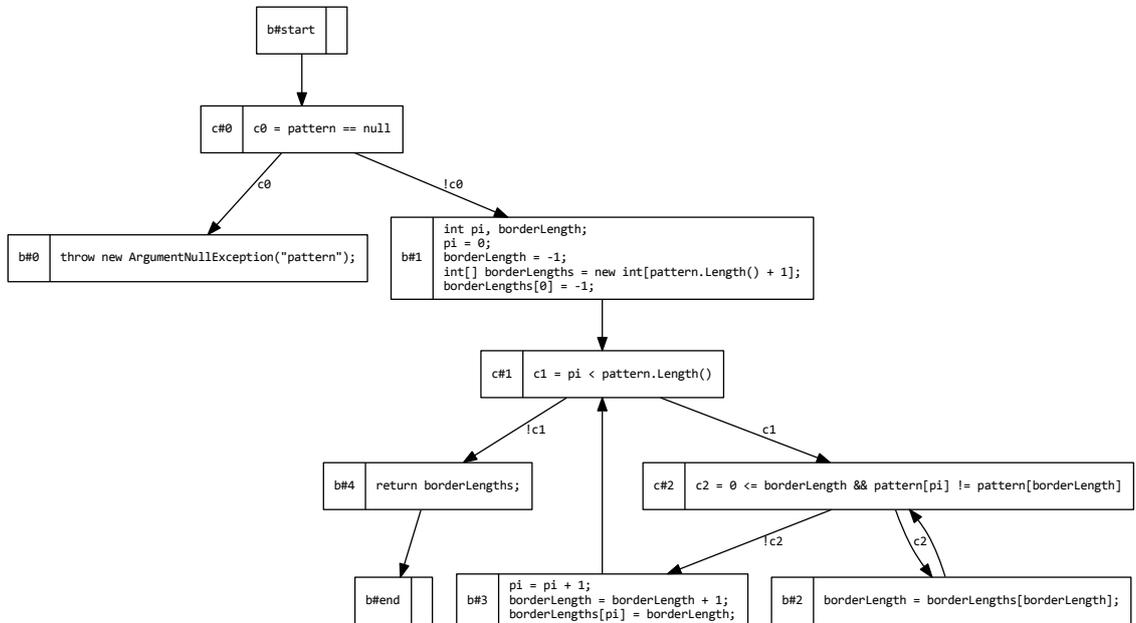


Testing Exercise (Solution)

1. Control Flow Graph (CFG) for the method maximalBorderLengths



2. Branch coverage

(a) Tests:

- null → ArgumentException
- "ab" → {-1, 0, 0}

(b) Minimal number of tests: 2

(c) No, statement coverage also needs at least two tests.

3. Loop coverage

(a) Tests:

- "" → {-1}: outer (0), inner (0, not reached)
- "a" → {-1, 0}: outer (1), inner (0)
- "ab" → {-1, 0, 0}: outer (2), inner (1)
- "aab" → {-1, 0, 1, 0}: outer (3), inner (2)

4. DU-Pair coverage

(a) DU-pairs for borderLength (bL for short)

i. Reaching Definitions

n	Reach(n)	ReachOut(n)
b#start	{}	{}
c#0	{}	{}
b#0	{}	{}
b#1	{}	{bL _{b#1} }
c#1	{bL _{b#1} , bL _{b#3} }	{bL _{b#1} , bL _{b#3} }
b#4	{bL _{b#1} , bL _{b#3} }	{bL _{b#1} , bL _{b#3} }
b#end	{bL _{b#1} , bL _{b#3} }	{bL _{b#1} , bL _{b#3} }
c#2	{bL _{b#1} , bL _{b#2} , bL _{b#3} }	{bL _{b#1} , bL _{b#2} , bL _{b#3} }
b#2	{bL _{b#1} , bL _{b#2} , bL _{b#3} }	{bL _{b#2} }
b#3	{bL _{b#1} , bL _{b#2} , bL _{b#3} }	{bL _{b#3} }

ii. DU-pairs

- (b#1, c#2), (b#1, b#2), (b#1, b#3)
- (b#2, c#2), (b#2, b#2), (b#2, b#3)
- (b#3, c#2), (b#3, b#2), (b#3, b#3)

(b) Tests:

- “aab” → {-1, 0, 1, 0} [c#0, b#1, c#1, c#2, b#3, c#1, c#2, b#3, c#1, c#2, b#2, c#2, b#2, c#2, b#3, c#1 b#4]

⇒ 8/9 DU-pairs covered = 88.9% DU-pair coverage

(c) Infeasible DU-pair: (b#1, b#2)

5. Bug detection

- Branch coverage: not detected
- Loop coverage: detected (because the inner loop needs to be executed more than once in a row)
- DU-Pair coverage: detected (because b#2 needs to be executed twice in a row)