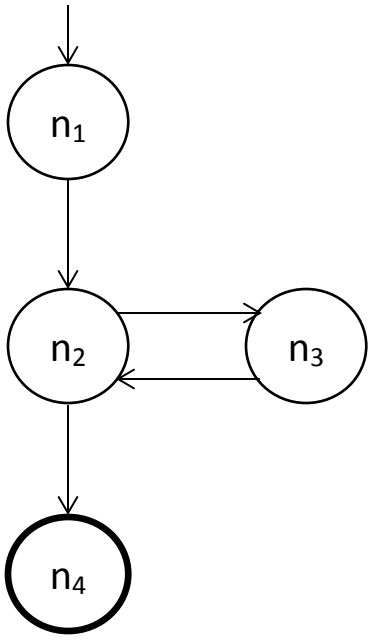
Name: Gabe Eapen
EID: eapengp
Class: EE382V (Software Testing Class)
HW#1 – Due Sep 16, 2014 @ 11:59pm

Exercises – Section 2.2.1 Question 3
Exercises – Section 2.2.1 Question 4
Exercises – Section 2.2.1 Question 5
Exercises – Section 2.2.1 Question 6

Q3) If we just considered the simple case of a many-to-one mapping that exists between the two criteria, then it would be clear that C_{strong} subsumes C_{weak}. However, by definition, proper criteria subsumption requires that every test set that that satisfies C_{strong} must also satisfy C_{weak}.
The question further states that it is not necessarily the case that T_{weak} is a subset of T_{strong}. How is this possible if C_{strong} subsumes C_{weak}? Well, that scenario is possible if there are infeasible test requirements in C_{strong} that are feasible in C_{weak}. Therefore in such a scenario, it is possible that the test set T_{weak} may reveal a fault that T_{strong} does not. This can also occur when touring with side trips can't avoid taking a side trip to meet a test requirement.

Q4)
a. Draw the graph



b. List test paths that achieve node coverage, but not edge coverage
It is not possible to have test paths that achieve node coverage but not edge coverage. The reason is that all test paths have to begin at Node 1 and when Node 3 is visited, the only way to get to Node 4 is going through Node 2. This then provides [2,3] and [3,2] edge coverage in addition to [1,2] and [2,4].

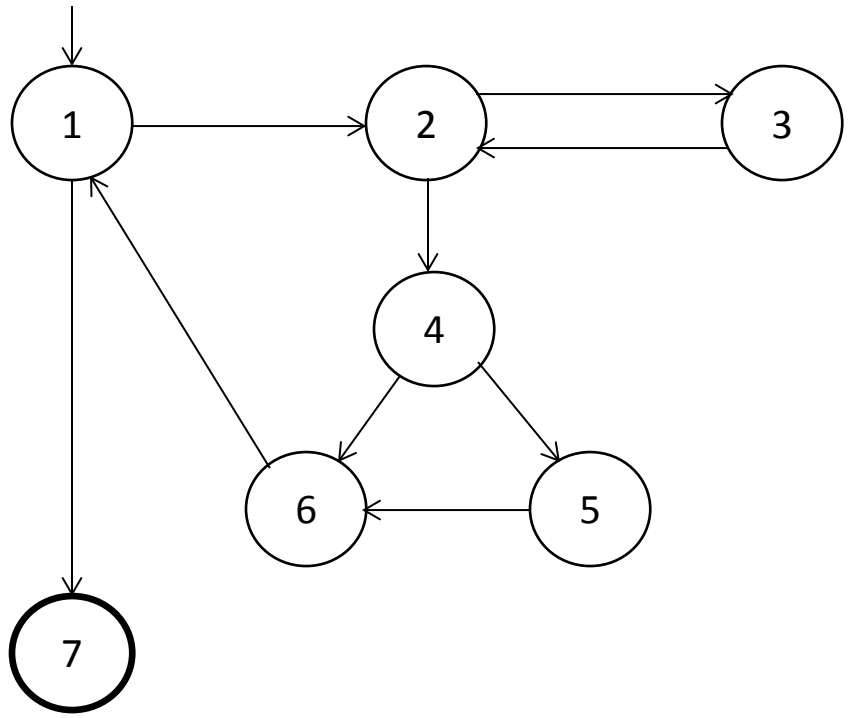
Therefore a test path ($t_0 = [n_1, n_2, n_3, n_2, n_4]$) that provides node coverage in this question **<u>also</u>** provides edge coverage.

c. List test paths that achieve edge coverage, but not edge-pair coverage
$t_0 = [n_1, n_2, n_3, n_2, n_4]$ achieves edge coverage but not edge-pair coverage as edge pair [3,2,3] is not toured

d. List test paths that achieve edge pair coverage
$t_0 = [n_1, n_2, n_3, n_2, n_3, n_2, n_4]$ ← tour with side trip
$t_1 = [n_1, n_2, n_4]$ ← tour with no side trip

$T = \{t_0, t_1\}$ is a test set that achieves edge-pair coverage

5)
a) Draw the graph



b) List the test requirements for edge-pair coverage
TR = {[1, 2, 3], [1, 2, 4], [2, 3, 2], [2, 4, 5], [2, 4, 6], [3, 2, 3], [3, 2, 4], [4, 5, 6], [4, 6, 1], [5, 6, 1], [6, 1, 7], [6, 1, 2]}

c) Does the given set of test paths satisfy edge-pair coverage? If not identify what is missing.
No, the test paths do not satisfy edge=pair coverage. The following test requirements are missing:
[3, 2, 3], [6, 1, 2]

d) Does the test path tour the simple path directly? With a side trip? If so identify the side trip.
The test path does not tour the simple path directly

The test path tours the simple path with a side trip of **[4, 6, 1, 2, 4]**

e) List the test requirements for node coverage, edge coverage, and prime path coverage.

Node Coverage:

TR = {1, 2, 3, 4, 5, 6, 7}

Edge Coverage:

TR = {(1, 2), (1, 7), (2, 3), (2, 4), (3, 2), (4, 5), (4, 6), (5, 6), (6, 1) }

Prime Path Coverage:

TR = {[2, 3, 2], [3, 2, 3], [1, 2, 4, 6, 1], [2, 4, 6, 1, 2], [4, 6, 1, 2, 3], [4, 6, 1, 2, 4], [6, 1, 2, 4, 6], [1, 2, 4, 5, 6, 1], [2, 4, 5, 6, 1, 2], [3, 2, 4, 6, 1, 7], [4, 5, 6, 1, 2, 3], [4, 5, 6, 1, 2, 4], [5, 6, 1, 2, 4, 5], [6, 1, 2, 4, 5, 6], [3, 2, 4, 5, 6, 1, 7]}

Work:

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

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

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

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

~~[1, 2, 4, 5, 6]~~
[1, 2, 4, 6, 1]*
~~[2, 4, 5, 6, 1]~~

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

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

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

f) List test paths that achieve node coverage but not edge coverage on the graph
T = {[1,2,3,2,4,5,6,1,7]}

g) List test paths that achieve edge coverage but not prime path coverage on the graph
path(t_1) = [1, 2, 4, 5, 6, 1, 7]
path(t_2) = [1, 2, 3, 2, 4, 6, 1, 7]
path(t_3) = [1, 2, 3, 2, 4, 5, 6, 1, 7]
path(t_4) = [1, 2, 4, 6, 1, 7]

T_1 = {t_1, t_2}
T_2 = {t_3, t_4}

6)
a. Enumerate the test requirements for node coverage, edge coverage, and prime path coverage on the graph
Node Coverage:
TR = {n_0, n_1, n_2, n_3, n_4, n_5, n_6, n_7, n_8, n_9}
Edge Coverage:
TR = {(n_0, n_3), (n_0, n_4), (n_1, n_4), (n_2, n_5), (n_2, n_6), (n_3, n_7), (n_4, n_7), (n_4, n_8), (n_5, n_1), (n_5, n_9), (n_6, n_9), (n_8, n_5)}
Prime Path Coverage:
TR = { [n0, n3, n7], [n0, n4, n7], [n0, n4, n8, n5, n1], [n0, n4, n8, n5, n9], [n1, n4, n8, n5, n1],[n1, n4, n8, n5, n9], [n2, n5, n1, n4, n7], [n2, n5, n1, n4, n8], [n2, n5, n9], [n2, n6, n9], [n4, n8, n5, n1, n4], [n5, n1, n4, n8, n5], [n8, n5, n1, n4, n7], [n8, n5, n1, n4, n8]}

b. List test paths that achieve node coverage but not edge coverage
T = {[n0, n3, n7], [n1, n4, n8, n5, n9], [n2, n6, n9]}

c. List test paths that achieve edge coverage but not prime path coverage.
T = {[n0, n3, n7], [n0, n4, n7], [n1, n4, n8, n5, n1, n4, n8], [n2, n6, n9], [n2, n5, n9]}