Faculty of Engineering, Architecture and Science
Department of Electrical and Computer Engineering
Laboratory Report Cover Page

Course Number	COE891
Course Title	Software Testing & QA
Semester/Year	Winter 2023
Instructor	Reza Semavi
TA Name	Hamed Karimi

Lab/Tutorial Report No.	6

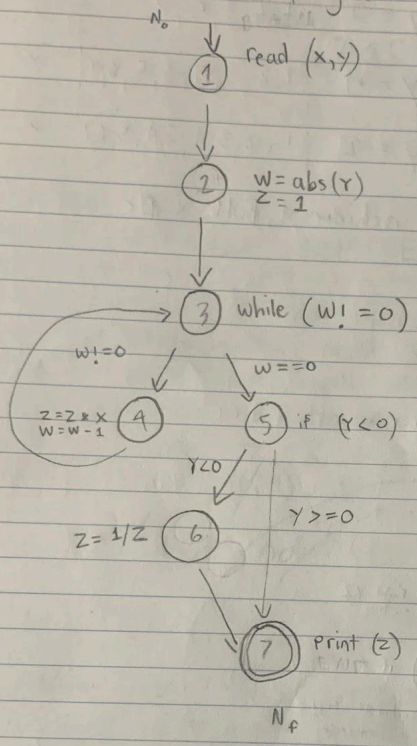
Section No.	012
Group No.	N/A
Submission Date	Mar 21th, 2023
Due Date	Mar 20th, 2023

Student Name	Student ID	Signature
Hamza Iqbal	500973673	H.I

Hamza Iqbal
500973673

COE 891

Q1. CFG of program



1. No infeasible paths since all possible paths can be explored

	values		test path	NC
2. i)	x = 1	y = 1	1,2,3,4,5,6,7	X
ii)	x = 0	y = 0	1,2,3,5,7	X
iii)	x = -1	y = -1	1,2,3,4,3,5,7	✓
iv)	x = 1	y = -2	1,2,3,4,3,4,3,5,6,7	✓

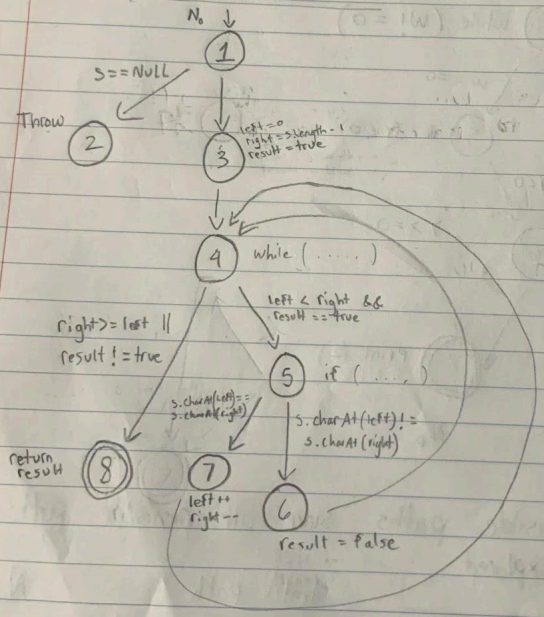
Negative values for y achieve full NC

	values	test path	EC
3. i)	x = 1, y = 1	1, 2, 3, 4, 3, 5, 7	✗
ii)	x = 0, y = 0	1, 2, 3, 5, 7	✗
iii)	x = -1, y = -1	1, 2, 3, 4, 3, 5, 6, 7	✓
iv)	x = 1, y = -2	1, 2, 3, 4, 3, 4, 5, 6, 7	✓

y < 0 achieves full EC

Q2.

1. CFG



2. TR (NC) = {[1], [2], [3], [4], [5], [6], [7], [8]}

TR (EC) = {[1,2], [1,3], [3,4], [4,5], [4,8], [5,6], [5,7], [6,4], [7,4]}

TR (EPC) = $\{[1,3,4],[3,4,5],[3,4,8],[4,5,6]$
$[4,5,7],[5,6,4],[5,7,4],[6,4,5],[6,4,8]$
$[7,4,5],[7,4,8]\}$

3.

NC But Not EC:

This is not possible, as all test paths must begin at node 1. A "null" case will traverse node 2, but not the rest. In order to traverse every other node, including node 7 & 6, will also hit edge (7,4) or (6,4).

EC but not EPC:

- $TS_1 = \{[1,2],[1,3,4,5,6,4,5,7,4,8]\}$

Note, in this set the EPC set:
$\{[7,4,5],[6,4,8],[3,4,8]\}$
are not touched by the test set, yet the set achieves EC.

Other TS:

- $TS_2 = \{[1,2],[1,3,4,5,7,4,5,6,4,8]\}$

Not EPC, missing: $\{[6,4,5],[7,4,8],[3,4,8]\}$

EPC

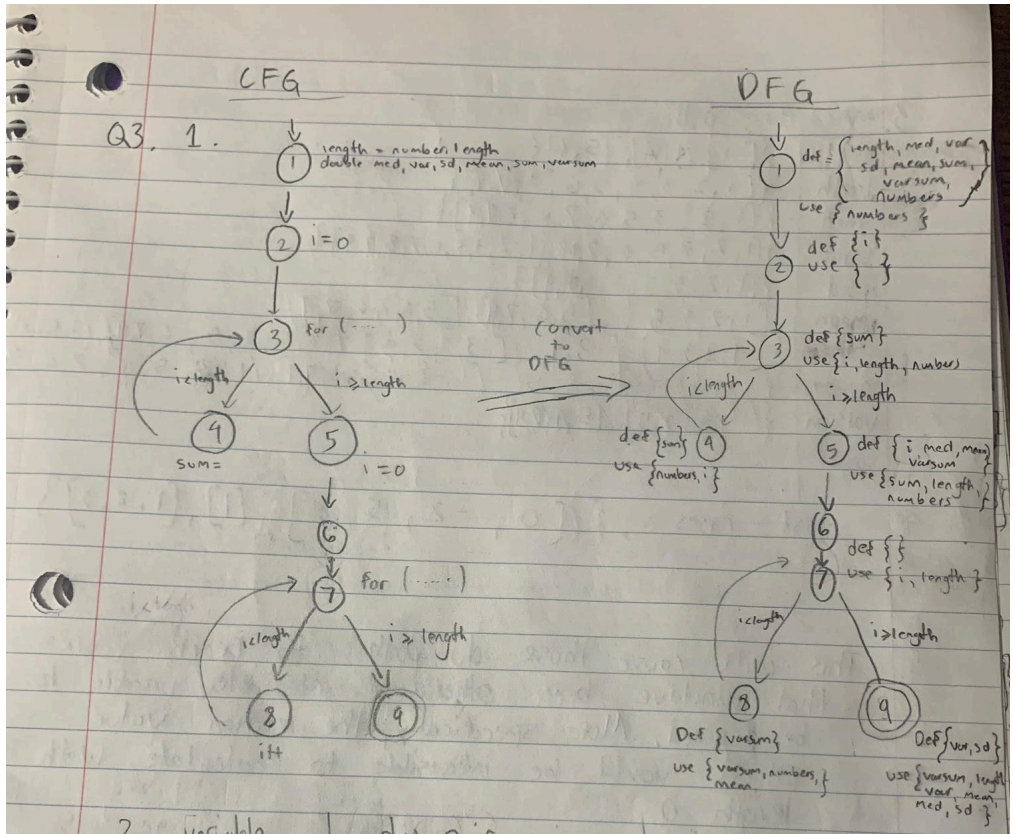
TS: {[1,3,4,5,6,4,5,7,4,8], [1,3,4,5,7,4,5,6,4,8], [1,3,4,8]}

4. TR (PPC): {[1,2], [1,3,4,8], [1,3,4,5,6], [1,3,4,5,7], [4,5,7,4], [4,5,6,4]}

There are no infeasible requirements as all TR can be satisfied.

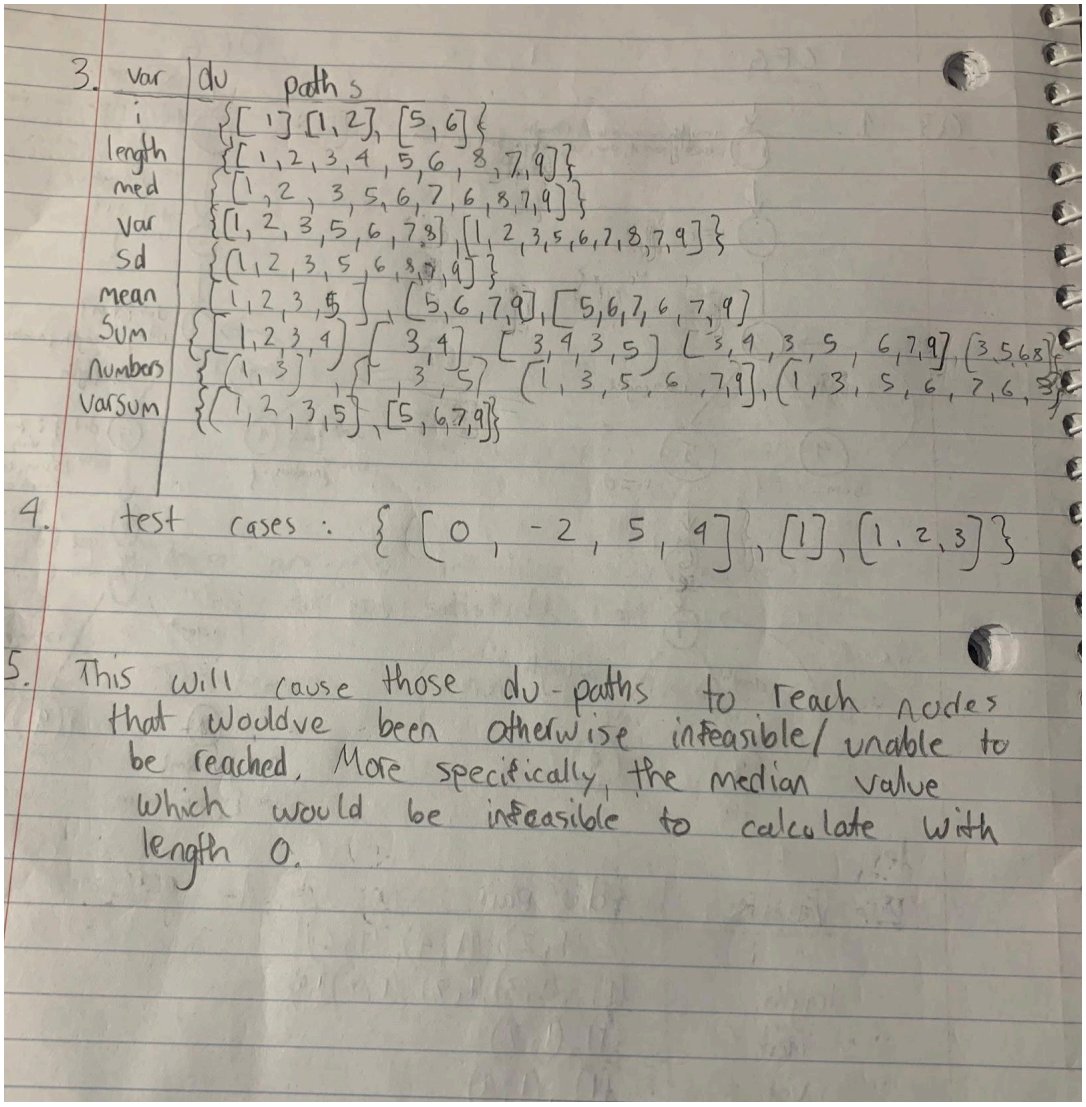
Test cases: { null, "", "racecar", "hello" }

Q3. 1.



2.

variable	du pair
i	(1,2), (1,4), (1,6)
length	(1), (1,3), (1,7), (1,8), (1,9)
med	(1), (1,7)
var	(1), (1,9)
sd	(1)
mean	(1), (1,7), (1,8)
sum	(1), (1,3), (1,4), (1,7)
numbers	(1,3), (1,4), (1,7), (1,8)
varsum	(1), (1,7), (1,8)


3.
var | du paths
i
length
med
var
sd
mean
sum
numbers
varsum
4. test cases :
5. This will cause those du-paths to reach nodes that wouldve been otherwise infeasible/ unable to be reached. More specifically, the median value which would be infeasible to calculate with length 0.

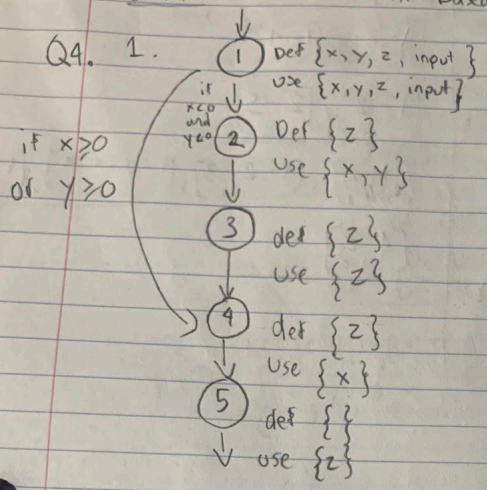
Q4.

DFG

overall

* Based nodes off of block numbers

Q4. 1.



2. def (1) = {x, y, z}
use (1) = {z, x, y}

def (2) = {z}
use (2) = {x, y}

def (3) = {z}
use (3) = {z}

def (4) = {z}
use (4) = {x}

def (5) = { }
use (5) = {z}

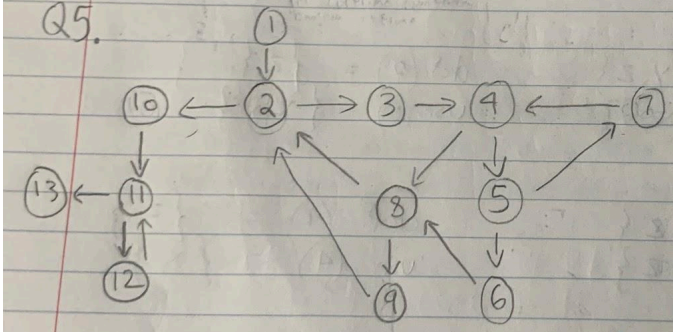
3.

node	P-use	C-use
1	x, y	z
2	y	z, x
3	-	z
4	-	z, x
5	-	-

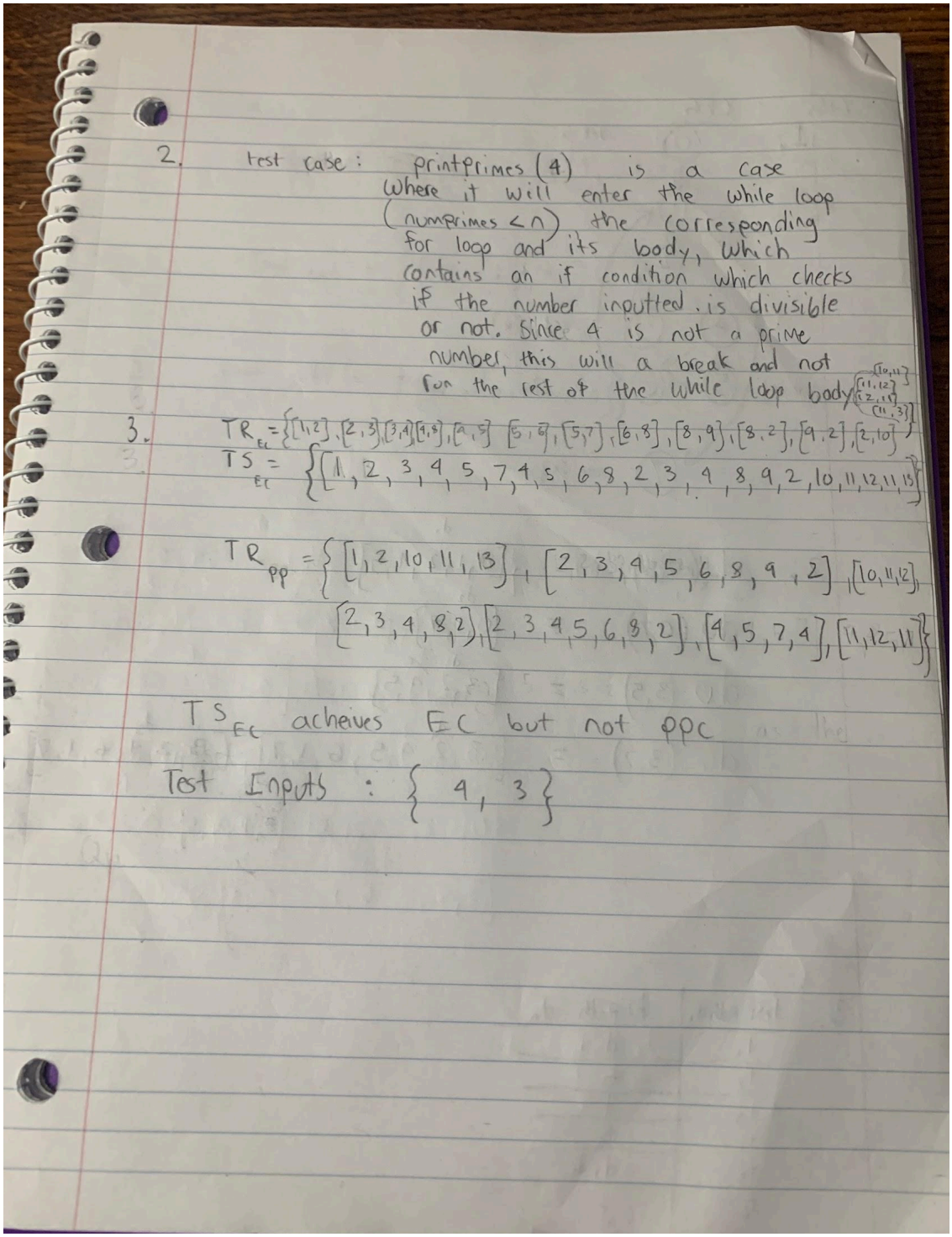
4.

Node	dc-paths	du-paths
1	All	[1-2], [1-4]
2	All	[1-2], [2-5]
3	All	[1-2], [1-2-3], [1-2-3-9] [1-4]
4	All	[4,5]
5	All	–

Q5.



1 : lines 4-12
2 : lines 13
3 : lines 14-15 + init
4 : line 16
5 : lines 17-18
6 : lines 19-21
7 : i++
8 : line 23
9 : line 24-25
10 : for loop init (23)
11 : i < numPrimes - 1
12 : line 29 + i++



2. Test case: printPrimes(4) is a case where it will enter the while loop (numPrimes < n) the corresponding for loop and its body, which contains an if condition which checks if the number inputted is divisible or not. Since 4 is not a prime number, this will a break and not run the rest of the while loop body

3. $TR_{EC} = \{[1,2],[2,3],[3,4],[4,5],[5,6],[5,7],[6,8],[8,9],[8,2],[9,2],[2,10]\}$

$TS_{EC} = \{[1,2,3,4,5,7,4,5,6,8,2,3,4,8,9,2,10,11,12,11,13]\}$

$TR_{pp} = \{[1,2,10,11,13],[2,3,4,5,6,8,9,2],[10,11,12],[2,3,4,8,2],[2,3,4,5,6,8,2],[4,5,7,4],[11,12,11]\}$

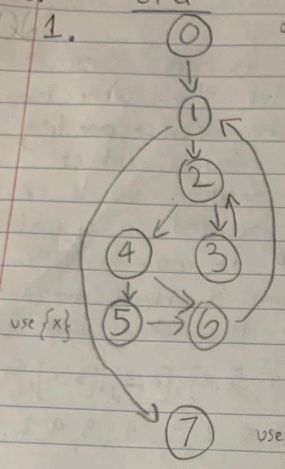
TS_{EC} acheives EC but not ppc

Test Inputs : { 4, 3 }

Q6.

Q6.

1. CFG



2. $du_x(0,5) = [0,1,2,4,5]$ (d_1)

$du_x(0,7) = [0,1,7]$ (d_2)

$du(3,5) = [3,2,4,5]$ (d_3)

$du(3,7) = [3,2,4,5,6,1,7], [3,2,4,6,1,7]$ (d_4, d_5)

3.

test path t_n	du path d_n
t_1	d_2
t_2	-
t_3	d_1
t_4	d_5
t_5	d_3 d_4
t_6	-

4. Test set = $\{[t_1, t_4], [t_1, t_5], [t_3, t_4], [t_3, t_5]\}$

5. Test set = $\{[t_1, t_3, t_5]\}$

6. test set = $\{[t_1, t_3, t_4, t_5]\}$