EECS Assignment 1

Q4.

A) CFG picture attached as requested.

B) Test Requirements for each coverage type:

1) Node Coverage:
   TR(NC) = {1,2,3,4,5,6,7,8,9,10,11}
2) Edge Coverage:
   TR(EC) = {[1,3], [1,2], [2,3], [3,4],[3,5],[3,6],[3,7],[4,5],[5,8],[6,7],[7,8], [8,9], [8,10], [9,11], [10,11] }
3) Edge-Pair Coverage:
   TR(EPC)={[1,2,3],[1,3,4],[1,3,5],[1,3,6],[1,3,7],[2,3,4],[2,3,5],[2,3,6],[2,3,7],[3,4,5],[3,5,8],[3,6,7],[3,7,8],[4,5,8],[5,8,9],[5,8,10],[6,7,8], [7,8,9], [7,8,10],[8,9,11],[8,10,11] }

   Infeasible paths = [5,8,10] and [7,8,10] are not possible paths as their conditions are contradictory that lead to them.

4) Prime-Path Coverage:

TR(PPC) =

Len 0	Len 1	Len 2	Len 3	Len 4	Len 5	Len 6	Len 7
1	1,2	1,2,3	1,2,3,4	1,2,3,4,5	1,2,3,4,5,8	1,2,3,4,5,8,9	[1,2,3,4,5,8,9,11]!
2	1,3	1,3,4	1,2,3,5	1,2,3,5,8	1,2,3,5,8,9	1,2,3,4,5,8,10	[1,2,3,4,5,8,10,11]!
3	2,3	1,3,5	1,2,3,6	1,2,3,6,7	1,2,3,5,8,10	[1,2,3,5,8,9,11]!	[1,2,3,6,7,8,9,11]!
4	3,4	1,3,6	1,2,3,7	1,2,3,7,8	1,2,3,6,7,8	[1,2,3,5,8,10,11]!	[1,2,3,6,7,8,10,11]!
5	3,5	1,3,7	2,3,4,5	1,3,6,7,8	1,2,3,7,8,9	[1,2,3,7,8,9,11]!	
6	3,6	2,3,4	2,3,5,8	1,3,7,8,9	1,2,3,7,8,10	[1,2,3,7,8,10,11]!	
7	3,7	2,3,5	2,3,6,7	1,3,7,8,10	1,3,4,5,8,9	1,2,3,6,7,8,9	
8	4,5	2,3,6	2,3,7,8	1,3,4,5,8	1,3,4,5,8,10	1,2,3,6,7,8,10	
9	5,8	2,3,7	3,4,5,8	1,3,5,8,9	1,3,6,7,8,9	[1,3,6,7,8,9,11]!	
10	6,7	3,4,5	3,5,8,9	1,3,5,8,10	1,3,6,7,8,10	[1,3,6,7,8,10,11]!	
[11]!	7,8	3,5,8	3,6,7,8	2,3,4,5,8	[1,3,5,8,9,11]!	[1,3,4,5,8,9,11]!	
	8,9	3,6,7	3,7,8,9	2,3,5,8,9	[1,3,5,8,10,11]!	[1,3,4,5,8,10,11]!	
	8,10	3,7,8	3,5,8,10	2,3,7,8,9	[1,3,7,8,9,11]!		
	[9,11]!	4,5,8	3,7,8,10	2,3,7,8,10	[1,3,7,8,10,11]!		
	[10,11]!	5,8,9	4,5,8,9	2,3,5,8,10	[2,3,5,8,9,11]!		
		5,8,10	4,5,8,10	2,3,6,7,8	[2,3,5,8,10,11]!		
		6,7,8	[5,8,9,11]!	3,4,5,8,9	2,3,4,5,8,9		
		7,8,9	[5,8,10,11]!	3,4,5,8,10	2,3,4,5,8,10		Prime Paths
		7,8,10	6,7,8,9	[3,5,8,9,11]!	2,3,6,7,8,9		
		[8,9,11]!	6,7,8,10	[3,5,8,10,11]!	[2,3,7,8,9,11]!		
		[8,10,11]!	[7,8,9,11]!	3,6,7,8,9	[2,3,7,8,10,11]!		
			[7,8,10,11]!	3,6,7,8,10	2,3,6,7,8,10		
				[3,7,8,10,11]!	[3,6,7,8,9,11]!		
				[3,7,8,9,11]!	[3,6,7,8,10,11]!		
				[4,5,8,9,11]!	[3,4,5,8,9,11]!		
				[4,5,8,10,11]!	[3,4,5,8,10,11]!		
				[6,7,8,10,11]!			
				[6,7,8,9,11]!			

Prime paths are colored red in the picture above and followed the procedure as shown in class. We have a total of 16 prime paths.

There are some infeasible prime test paths as well in the code:

1) [1,3,7,8,10,11] (7 and 10 cannot be in same path)
2) [1,3,5,8,10,11] (5 and 10 cannot be in same path)
3) [1,3,4,5,8,9,11] (4 and 9 cannot be in same path)
4) [1,3,6,7,8,10,11] (6,7 and 10 cannot be in same path)
5) [1,2,3,7,8,10,11] (7 and 10 cannot be in same path)
6) [1,2,3,5,8,10,11] (5 and 10 cannot be in same path)
7) [1,2,3,6,7,8,10,11] (6,7 and 10 cannot be in same path)
8) [1,2,3,4,5,8,9,11] (4 and 9 cannot be in same path)

The conditions in node 4 ,9 and node 5,6,7,10 contradict each other preceding nodes. If arg.length() == 0 then it cannot be true for 8 therefore something that goes from 4 cannot pass through 9. And similarly for other parts.