CS206 Project Report

Priyanka Pothala
ppoth001@ucr.edu

Teja Vemparala
tvemp001@ucr.edu

1. Experimental Results

Benchmark	Coverage Criteria	Prioritization Method	# testcases	# faults exposed
printtokens	Statement	Random	18	3
		Total	6	6
		Additional	5	4
	Branch	Random	16	4
		Total	7	5
		Additional	6	4
printtokens2	Statement	Random	11	6
		Total	4	7
		Additional	4	6
	Branch	Random	18	7
		Total	6	8
		Additional	4	7
replace	Statement	Random	22	10
		Total	13	10
		Additional	9	4
	Branch	Random	27	17
		Total	21	19
		Additional	11	13
schedule	Statement	Random	6	2
		Total	3	2
		Additional	3	4
	Branch	Random	10	7
		Total	8	3
		Additional	7	5
schedule2	Statement	Random	3	1
		Total	1	3
		Additional	1	3
	Branch	Random	11	5
		Total	7	5
		Additional	5	2
tcas	Statement	Random	5	7
		Total	4	10
		Additional	4	10

	Branch	Random	12	17
		Total	13	13
		Additional	11	13
totinfo	Statement	Random	9	13
		Total	5	12
		Additional	5	12
	Branch	Random	7	11
		Total	5	13
		Additional	5	13

2. Observations

- The generated test suites are very small when compared to the original number of available test cases
- Statement coverage generated the smallest test suite and Additional coverage prioritization generated the smallest test suite for both statement and branch coverage.
- Total coverage prioritization exposed more faults than random and additional coverage prioritization methods
- Random prioritization method was not very consistent in exposing the faults i.e. sometimes it performed better than total and additional coverage prioritization and sometimes it wasn’t
- Branch coverage was more effective than statement coverage in exposing the faults. However, none of the coverage criteria and prioritization methods were able to expose all the faults