

Software testing without the oracle correctness assumption

Tun LI, Wanwei LIU, Xinrui GUO, Ji WANG

Frontiers of Computer Science, DOI: [10.1007/s11704-019-8434-4](https://doi.org/10.1007/s11704-019-8434-4)

Problems & Ideas

- When testing programs, the oracle correctness assumption (OCA) implies that there are no errors in the test oracles, such as the expected outputs are always correctly designed and written in unit testing assertions.
- Without OCA, what will the testing be?
- Main idea:
 - to estimate the number of tests applied to a program under testing (PUT) to reveal the errors in oracles

Table 1 The Motivating Example [1]

Function: sort, Input: (a, b, c)	(1,2,3)	(1,3,2)	(2,1,3)	(2,3,1)	(3,1,2)	(3,2,1)
1 if(a > b){	•	•	•	•	•	•
2 swap(a, b);}			•		•	•
3 if(b > c){	•	•	•	•	•	•
4 swap(b, c);		•		•	•	•
5 if(a < b){ //should be if(a > b)		•		•	•	•
6 swap(a, b);		•			•	
7 }						
Expected Output (correct)	(1,2,3)✓	(1,2,3)×	(1,2,3)✓	(1,2,3)×	(1,2,3)×	(1,2,3)×
Expected Output (incorrect)	≠(1,2,3)×	(2,1,3)✓	≠(1,2,3)×	(2,1,3)✓	(2,1,3)✓	(2,1,3)✓
Actual Output	(1,2,3)	(2,1,3)	(1,2,3)	(2,1,3)	(2,1,3)	(2,1,3)

Main Contributions

- The testing without OCA for unit testing is discussed systematically in terms of a mathematical model;
- The expected number of testing times which are sufficient to reveal errors in oracles is proposed;
- Some existing conclusions on software testing are explained from a new view by the mathematical model.