

Michael Pradel

List of Publications by Year in descending order

Source: <https://exaly.com/author-pdf/11431750/publications.pdf>

Version: 2024-02-01

27
papers

1,093
citations

1937685

4
h-index

2053705

5
g-index

27
all docs

27
docs citations

27
times ranked

471
citing authors

#	ARTICLE	IF	CITATIONS
1	Automated program repair. Communications of the ACM, 2019, 62, 56-65.	4.5	193
2	Automatic Generation of Object Usage Specifications from Large Method Traces. , 2009, , .		104
3	Performance issues and optimizations in JavaScript. , 2016, , .		90
4	A framework for the evaluation of specification miners based on finite state machines. , 2010, , .		69
5	Fully automatic and precise detection of thread safety violations. , 2012, , .		61
6	Performance regression testing of concurrent classes. , 2014, , .		60
7	JITProf: pinpointing JIT-unfriendly JavaScript code. , 2015, , .		43
8	Statically checking API protocol conformance with mined multi-object specifications. , 2012, , .		42
9	Ballerina: Automatic generation and clustering of efficient random unit tests for multithreaded code. , 2012, , .		42
10	Nomen est omen. , 2016, , .		42
11	Performance problems you can fix: a dynamic analysis of memoization opportunities. , 2015, , .		41
12	SyncProf: detecting, localizing, and optimizing synchronization bottlenecks. , 2016, , .		33
13	EventBreak. , 2014, , .		32
14	Efficient Detection of Thread Safety Violations via Coverage-Guided Generation of Concurrent Tests. , 2017, , .		32
15	Monkey see, monkey do: effective generation of GUI tests with inferred macro events. , 2016, , .		31
16	Leveraging test generation and specification mining for automated bug detection without false positives. , 2012, , .		30
17	Detecting argument selection defects. , 2017, 1, 1-22.		29
18	BitA: Coverage-guided, automatic testing of actor programs. , 2013, , .		17

#	ARTICLE	IF	CITATIONS
19	Pinpointing and repairing performance bottlenecks in concurrent programs. Empirical Software Engineering, 2018, 23, 3034-3071.	3.9	17
20	Name-Based Analysis of Equally Typed Method Arguments. IEEE Transactions on Software Engineering, 2013, 39, 1127-1143.	5.6	15
21	Performance problems you can fix: a dynamic analysis of memoization opportunities. ACM SIGPLAN Notices, 2015, 50, 607-622.	0.2	15
22	Scaffle: bug localization on millions of files. , 2020, , .		15
23	Automatic testing of sequential and concurrent substitutability. , 2013, , .		10
24	Bugs in Quantum computing platforms: an empirical study. , 2022, 6, 1-27.		10
25	An actionable performance profiler for optimizing the order of evaluations. , 2017, , .		9
26	No strings attached. , 2020, , .		9
27	Systematic black-box analysis of collaborative web applications. ACM SIGPLAN Notices, 2017, 52, 171-184.	0.2	2