

Jens Krinke

List of Publications by Year in descending order

Source: <https://exaly.com/author-pdf/7664379/publications.pdf>

Version: 2024-02-01

81
papers

2,549
citations

567281

15
h-index

454955

30
g-index

83
all docs

83
docs citations

83
times ranked

1073
citing authors

#	ARTICLE	IF	CITATIONS
1	Comparison and Evaluation of Clone Detection Tools. IEEE Transactions on Software Engineering, 2007, 33, 577-591.	5.6	567
2	A Study of Consistent and Inconsistent Changes to Code Clones. Reverse Engineering (WCRE), Working Conference on, 2007, , .	0.0	117
3	Efficient path conditions in dependence graphs for software safety analysis. ACM Transactions on Software Engineering and Methodology, 2006, 15, 410-457.	6.0	113
4	Searching for better configurations: a rigorous approach to clone evaluation. , 2013, , .		109
5	Is Cloned Code More Stable than Non-cloned Code?. , 2008, , .		103
6	Static slicing of threaded programs. , 1998, , .		84
7	A comparison of code similarity analysers. Empirical Software Engineering, 2018, 23, 2464-2519.	3.9	84
8	Deep Parameter Optimisation. , 2015, , .		74
9	ORBS: language-independent program slicing. , 2014, , .		57
10	Aspect mining using event traces. , 0, , .		53
11	Comparative stability of cloned and non-cloned code. , 2012, , .		51
12	Context-sensitive slicing of concurrent programs. , 2003, , .		46
13	Is cloned code older than non-cloned code?. , 2011, , .		46
14	Toxic Code Snippets on Stack Overflow. IEEE Transactions on Software Engineering, 2021, 47, 560-581.	5.6	46
15	Dependence clusters in source code. ACM Transactions on Programming Languages and Systems, 2009, 32, 1-33.	2.1	39
16	Siamese: scalable and incremental code clone search via multiple code representations. Empirical Software Engineering, 2019, 24, 2236-2284.	3.9	38
17	Static slicing of threaded programs. ACM SIGPLAN Notices, 1998, 33, 35-42.	0.2	35
18	Search based data sensitivity analysis applied to requirement engineering. , 2009, , .		33

#	ARTICLE	IF	CITATIONS
19	State-based model slicing. ACM Computing Surveys, 2013, 45, 1-36.	23.0	33
20	Behind the Intents. , 2020, , .		33
21	Cloning and copying between GNOME projects. , 2010, , .		31
22	EzUnit: A Framework for Associating Failed Unit Tests with Potential Programming Errors. , 2007, , 101-104.		31
23	Validation of measurement software as an application of slicing and constraint solving. Information and Software Technology, 1998, 40, 661-675.	4.4	29
24	Empirical study of optimization techniques for massive slicing. ACM Transactions on Programming Languages and Systems, 2007, 30, 3.	2.1	29
25	Exact scalable sensitivity analysis for the next release problem. ACM Transactions on Software Engineering and Methodology, 2014, 23, 1-31.	6.0	29
26	Using compilation/decompilation to enhance clone detection. , 2017, , .		28
27	Slicing, Chopping, and Path Conditions with Barriers. Software Quality Journal, 2004, 12, 339-360.	2.2	27
28	An Empirical Study of the Impacts of Clones in Software Maintenance. , 2011, , .		25
29	ORBS and the limits of static slicing. , 2015, , .		25
30	CROP. , 2018, , .		25
31	Similarity of Source Code in the Presence of Pervasive Modifications. , 2016, , .		23
32	An empirical study on dependence clusters for effort-aware fault-proneness prediction. , 2016, , .		23
33	Effects of context on program slicing. Journal of Systems and Software, 2006, 79, 1249-1260.	4.5	21
34	Visualization of program dependence and slices. , 0, , .		20
35	Are developers aware of the architectural impact of their changes?. , 2017, , .		20
36	A picture is worth a thousand words: Code clone detection based on image similarity. , 2018, , .		20

#	ARTICLE	IF	CITATIONS
37	Memory mutation testing. Information and Software Technology, 2017, 81, 97-111.	4.4	19
38	Advanced slicing of sequential and concurrent programs. , 0, , .		18
39	Software Engineering Projects in Distant Teaching. , 2005, , .		18
40	Pitfalls in Aspect Mining. , 2008, , .		17
41	Establishing multilevel test-to-code traceability links. , 2020, , .		17
42	Mining Control Flow Graphs for Crosscutting Concerns. , 2006, , .		16
43	Intransitive Noninterference in Dependence Graphs. , 2006, , .		16
44	Python Coding Style Compliance on Stack Overflow. , 2019, , .		16
45	Dynamic path conditions in dependence graphs. , 2006, , .		16
46	Generalized observational slicing for tree-represented modelling languages. , 2017, , .		14
47	Issues in clone classification for dataflow languages. , 2010, , .		13
48	A comparison of tree- and line-oriented observational slicing. Empirical Software Engineering, 2019, 24, 3077-3113.	3.9	13
49	The Impact of Code Review on Architectural Changes. IEEE Transactions on Software Engineering, 2021, 47, 1041-1059.	5.6	13
50	Unions of slices are not slices. , 0, , .		12
51	Distinguishing copies from originals in software clones. , 2010, , .		12
52	Ethics in the mining of software repositories. Empirical Software Engineering, 2022, 27, 1.	3.9	12
53	Coherent clusters in source code. Journal of Systems and Software, 2014, 88, 1-24.	4.5	10
54	Tree-Oriented vs. Line-Oriented Observation-Based Slicing. , 2017, , .		10

#	ARTICLE	IF	CITATIONS
55	Context-sensitivity matters, but context does not. , 0, , .		9
56	Sentinel: A Hyper-Heuristic for the Generation of Mutant Reduction Strategies. IEEE Transactions on Software Engineering, 2022, 48, 803-818.	5.6	9
57	Mutation testing of memory-related operators. , 2015, , .		8
58	Genetic Improvement using Higher Order Mutation. , 2015, , .		8
59	HOMI: Searching Higher Order Mutants for Software Improvement. Lecture Notes in Computer Science, 2016, , 18-33.	1.3	8
60	Does code review really remove coding convention violations?. , 2020, , .		8
61	Dependence cluster visualization. , 2010, , .		7
62	Ethical Mining. , 2020, , .		7
63	Coherent dependence clusters. , 2010, , .		6
64	AVPredictor: Comprehensive prediction and detection of atomicity violations. Concurrency Computation Practice and Experience, 2019, 31, e5160.	2.2	6
65	Statement-Level Cohesion Metrics and their Visualization. , 2007, , .		5
66	PROGRAM SLICING. , 2005, , 307-332.		4
67	Characterising, Explaining, and Exploiting the Approximate Nature of Static Analysis through Animation. , 2006, , .		4
68	TestNMT: function-to-test neural machine translation. , 2018, , .		4
69	Artefact Relation Graphs for Unit Test Reuse Recommendation. , 2021, , .		4
70	Who's this?. , 2018, , .		3
71	Evaluating lexical approximation of program dependence. Journal of Systems and Software, 2020, 160, 110459.	4.5	3
72	Pidgin Crasher: Searching for Minimised Crashing GUI Event Sequences. Lecture Notes in Computer Science, 2014, , 253-258.	1.3	3

#	ARTICLE	IF	CITATIONS
73	Mining execution relations for crosscutting concerns. IET Software, 2008, 2, 65.	2.1	2
74	Efficient Identification of Linchpin Vertices in Dependence Clusters. ACM Transactions on Programming Languages and Systems, 2013, 35, 1-35.	2.1	2
75	Code Similarity in Clone Detection. , 2021, , 135-150.		2
76	Searching for Configurations in Clone Evaluation – A Replication Study. Lecture Notes in Computer Science, 2016, , 250-256.	1.3	2
77	“No Good Reason to Remove Features” Lecture Notes in Computer Science, 2017, , 25-44.	1.3	2
78	TCTracer: Establishing test-to-code traceability links using dynamic and static techniques. Empirical Software Engineering, 2022, 27, 1.	3.9	2
79	MOBS. , 2018, , .		1
80	Special issue on mutation testing and analysis. Software Testing Verification and Reliability, 2019, 29, e1697.	2.0	1
81	Software-Sicherheitsprüfung mit VALSOFT. Computer Science - Research and Development, 1999, 14, 62-73.	0.9	0