Martin Wirsing

List of Publications by Year
in descending order

Source: https://exaly.com/author-pdf/11210926/publications.pdf

Version: 2024-02-01

45 papers	1,173 citations	516710 16 h-index	454955 30 g-index
47 all docs	47 docs citations	47 times ranked	236 citing authors

#	Article	IF	Citations
1	Algebraic Specification. , 1990, , 675-788.		170
2	Partial abstract types. Acta Informatica, 1982, 18, 47-64.	0.5	146
3	Structured algebraic specifications: A Kernel language. Theoretical Computer Science, 1986, 42, 123-249.	0.9	99
4	On hierarchies of abstract data types. Acta Informatica, 1983, 20, 1.	0.5	97
5	Behavioural and abstractor specifications. Science of Computer Programming, 1995, 25, 149-186.	1.9	71
6	On the algebraic definition of programming languages. ACM Transactions on Programming Languages and Systems, 1987, 9, 54-99.	2.1	52
7	Algebraic implementations preserve program correctness. Science of Computer Programming, 1986, 7, 35-53.	1.9	38
8	Engineering of Software-Intensive Systems: State of the Art and Research Challenges. Lecture Notes in Computer Science, 2008, , 1-44.	1.3	32
9	Proof systems for structured specifications with observability operators. Theoretical Computer Science, 1997, 173, 393-443.	0.9	29
10	Rigorous engineering of collective adaptive systems: special section. International Journal on Software Tools for Technology Transfer, 2020, 22, 389-397.	1.9	27
11	ASCENS: Engineering Autonomic Service-Component Ensembles. Lecture Notes in Computer Science, 2013, , 1-24.	1.3	27
12	The Ensemble Development Life Cycle and Best Practices for Collective AutonomicÂSystems. Lecture Notes in Computer Science, 2015, , 325-354.	1.3	24
13	A Formal Approach to Object-Oriented Software Engineering. Electronic Notes in Theoretical Computer Science, 1996, 4, 322-360.	0.9	22
14	A Component Model for Architectural Programming. Electronic Notes in Theoretical Computer Science, 2006, 160, 75-96.	0.9	22
15	A Life Cycle for the Development of Autonomic Systems: The E-mobility Showcase. , 2013, , .		19
16	Algebraic specifications of reachable higher-order algebras. Lecture Notes in Computer Science, 1988, , 154-169.	1.3	19
17	Continuous Abstract Data Types. Fundamenta Informaticae, 1986, 9, 95-125.	0.4	17
18	Algebraic specification languages: An overview. Lecture Notes in Computer Science, 1995, , 81-115.	1.3	16

#	Article	IF	Citations
19	Timed rewriting logic with an application to object-based specification. Science of Computer Programming, 1997, 28, 225-246.	1.9	16
20	A Rewriting Logic Framework for Soft Constraints. Electronic Notes in Theoretical Computer Science, 2007, 176, 181-197.	0.9	15
21	Collective Autonomic Systems: Towards Engineering Principles and Their Foundations. Lecture Notes in Computer Science, 2016, , 180-200.	1.3	15
22	Which Soft Constraints do you Prefer?. Electronic Notes in Theoretical Computer Science, 2009, 238, 189-205.	0.9	13
23	A modular framework for specification and implementation. Lecture Notes in Computer Science, 1989, , 42-73.	1.3	12
24	Dynamic Logic for Ensembles. Lecture Notes in Computer Science, 2018, , 32-47.	1.3	12
25	On the algebraic extensions of abstract data types. Lecture Notes in Computer Science, 1981, , 244-251.	1.3	11
26	Semantics, distributed implementation, and formal analysis of KLAIM models in Maude. Science of Computer Programming, 2015, 99, 24-74.	1.9	10
27	Algebraic specification with built-in domain constructions. Lecture Notes in Computer Science, 1988, , 132-148.	1.3	10
28	Algebraic definition of a functional programming language and its semantic models. RAIRO - Theoretical Informatics and Applications, 1983, 17, 137-161.	0.7	9
29	Modelling the CoCoME with the Java/A Component Model. Lecture Notes in Computer Science, 2008, , 207-237.	1.3	8
30	A heterogeneous approach to service-oriented systems specification. , 2010, , .		7
31	Correct Realizations of Interface Constraints with OCL. Lecture Notes in Computer Science, 1999, , 399-415.	1.3	7
32	Model-Checking Helena Ensembles withÂSpin. Lecture Notes in Computer Science, 2015, , 331-360.	1.3	7
33	Relational specifications and observational semantics. , 1986, , 209-217.		5
34	(Objects + concurrency) & reusability â€" A proposal to circumvent the inheritance anomaly. Lecture Notes in Computer Science, 1996, , 232-247.	1.3	5
35	A contract-based approach to adaptivity. The Journal of Logic and Algebraic Programming, 2011, 80, 180-193.	1.4	5
36	Rigorous Engineering of Collective Adaptive Systems Track Introduction. Lecture Notes in Computer Science, 2016, , 535-538.	1.3	5

#	Article	IF	Citations
37	Behavioural and abstractor specifications revisited. Theoretical Computer Science, 2018, 741, 32-43.	0.9	5
38	Introduction to â€œRigorous Engineering of Autonomic Ensemblesâ€â€“ Track Introduction. Lecture Notes in Computer Science, 2014, , 96-98.	1.3	5
39	Generalized heterogeneous algebras and partial interpretations. Lecture Notes in Computer Science, 1983, , 1-34.	1.3	4
40	Rigorous Engineering of Collective Adaptive Systems Introduction to the 2nd Track Edition. Lecture Notes in Computer Science, 2018, , 3-12.	1.3	4
41	Synthesizing safe policies under probabilistic constraints with reinforcement learning and Bayesian model checking. Science of Computer Programming, 2021, 206, 102620.	1.9	3
42	Bisimulation in Algebraic Specifications11Work partially supported by the ESPRIT-project METEOR and by CNR (Italy).. , 1989, , 1-31.		2
43	Continuous abstract data types: Basic machinery and results. , 1985, , 431-441.		1
44	Algebraic Reinforcement Learning. Lecture Notes in Computer Science, 2015, , 562-579.	1.3	1
45	Rigorous Engineering of Collective Adaptive Systems Introduction to the 3rd Track Edition. Lecture Notes in Computer Science, 2020, , 161-170.	1.3	1