

# Zhong Shao

## List of Publications by Year in descending order

Source: <https://exaly.com/author-pdf/1525663/publications.pdf>

Version: 2024-02-01

34  
papers

439  
citations

1307594

7  
h-index

996975

15  
g-index

34  
all docs

34  
docs citations

34  
times ranked

229  
citing authors

#	ARTICLE	IF	CITATIONS
1	Deep Specifications and Certified Abstraction Layers. , 2015, , .		102
2	Flexible representation analysis. ACM SIGPLAN Notices, 1997, 32, 85-98.	0.2	45
3	Toward compositional verification of interruptible OS kernels and device drivers. , 2016, , .		35
4	Certifying Low-Level Programs with Hardware Interrupts and Preemptive Threads. Journal of Automated Reasoning, 2009, 42, 301-347.	1.4	28
5	Deep Specifications and Certified Abstraction Layers. ACM SIGPLAN Notices, 2015, 50, 595-608.	0.2	26
6	A syntactic approach to foundational proof-carrying code. , 0, , .		24
7	A Syntactic Approach to Foundational Proof-Carrying Code. Journal of Automated Reasoning, 2003, 31, 191-229.	1.4	21
8	Position paper: the science of deep specification. Philosophical Transactions Series A, Mathematical, Physical, and Engineering Sciences, 2017, 375, 20160331.	3.4	21
9	Characterizing Progress Properties of Concurrent Objects via Contextual Refinements. Lecture Notes in Computer Science, 2013, , 227-241.	1.3	19
10	Certified concurrent abstraction layers. ACM SIGPLAN Notices, 2018, 53, 646-661.	0.2	11
11	Type-based amortized resource analysis with integers and arrays. Journal of Functional Programming, 2015, 25, .	0.8	10
12	Foundational Typed Assembly Language with Certified Garbage Collection. , 2007, , .		9
13	Typed cross-module compilation. ACM SIGPLAN Notices, 1999, 34, 141-152.	0.2	8
14	DeepSEA: a language for certified system software. , 2019, 3, 1-27.		8
15	A type-based compiler for standard ML. ACM SIGPLAN Notices, 1995, 30, 116-129.	0.2	7
16	Certifying low-level programs with hardware interrupts and preemptive threads. ACM SIGPLAN Notices, 2008, 43, 170-182.	0.2	7
17	A new hierarchical software architecture towards safety-critical aspects of a drone system. Frontiers of Information Technology and Electronic Engineering, 2019, 20, 353-362.	2.6	6
18	WormSpace. , 2019, , .		6

#	ARTICLE	IF	CITATIONS
19	A Simple Model for Certifying Assembly Programs with First-Class Function Pointers. , 2011, , .		5
20	Toward Compositional Verification of Interruptible OS Kernels and Device Drivers. Journal of Automated Reasoning, 2018, 61, 141-189.	1.4	5
21	Fully reflexive intensional type analysis. ACM SIGPLAN Notices, 2000, 35, 82-93.	0.2	5
22	Implementing typed intermediate languages. ACM SIGPLAN Notices, 1999, 34, 313-323.	0.2	4
23	Principled scavenging. ACM SIGPLAN Notices, 2001, 36, 81-91.	0.2	4
24	Inlining as staged computation. Journal of Functional Programming, 2003, 13, 647-676.	0.8	4
25	Toward compositional verification of interruptible OS kernels and device drivers. ACM SIGPLAN Notices, 2016, 51, 431-447.	0.2	4
26	Much ADO about failures: a fault-aware model for compositional verification of strongly consistent distributed systems. , 2021, 5, 1-31.		4
27	Proving the correctness of concurrent robot software. , 2012, , .		3
28	Transparent modules with fully syntatic signatures. ACM SIGPLAN Notices, 1999, 34, 220-232.	0.2	3
29	Trace-Based Temporal Verification for Message-Passing Programs. , 2014, , .		2
30	Adore: atomic distributed objects with certified reconfiguration. , 2022, , .		2
31	Weak Updates and Separation Logic. New Generation Computing, 2011, 29, 3-29.	3.3	1
32	Type-preserving compilation of featherweight java. , 0, , .		0
33	Modular Development of Certified System Software. , 2009, , .		0
34	Representing Java classes in a typed intermediate language. ACM SIGPLAN Notices, 1999, 34, 183-196.	0.2	0