

# Constance Heitmeyer Or Connie Heitmeyer

## List of Publications by Year in descending order

Source: <https://exaly.com/author-pdf/1409876/publications.pdf>

Version: 2024-02-01

32  
papers

1,345  
citations

933447

10  
h-index

713466

21  
g-index

35  
all docs

35  
docs citations

35  
times ranked

459  
citing authors

#	ARTICLE	IF	CITATIONS
1	Automated consistency checking of requirements specifications. ACM Transactions on Software Engineering and Methodology, 1996, 5, 231-261.	6.0	448
2	Using abstraction and model checking to detect safety violations in requirements specifications. IEEE Transactions on Software Engineering, 1998, 24, 927-948.	5.6	150
3	A security model for military message systems. ACM Transactions on Computer Systems, 1984, 2, 198-222.	0.8	135
4	Requirements Engineering and Technology Transfer: Obstacles, Incentives and Improvement Agenda. Requirements Engineering, 2002, 7, 113-123.	3.1	107
5	Model Checking Complete Requirements Specifications Using Abstraction. Automated Software Engineering, 1999, 6, 37-68.	2.9	92
6	Using Model Checking to Generate Tests from Requirements Specifications. Lecture Notes in Computer Science, 1999, , 146-162.	1.3	72
7	Applying Formal Methods to a Certifiably Secure Software System. IEEE Transactions on Software Engineering, 2008, 34, 82-98.	5.6	60
8	Automatic generation of state invariants from requirements specifications. , 1998, , .		56
9	Formal specification and verification of data separation in a separation kernel for an embedded system. , 2006, , .		47
10	Proving Invariants of I/O Automata with TAME. Automated Software Engineering, 2002, 9, 201-232.	2.9	23
11	Program Synthesis from Formal Requirements Specifications Using APTS. Higher-Order and Symbolic Computation, 2003, 16, 63-92.	0.3	20
12	Generating optimized code from SCR specifications. , 2006, , .		14
13	Building high assurance human-centric decision systems. Automated Software Engineering, 2015, 22, 159-197.	2.9	12
14	A strategy for efficiently verifying requirements. , 2003, , .		11
15	Obtaining Trust in Autonomous Systems: Tools for Formal Model Synthesis and Validation. , 2015, , .		10
16	Managing Complexity in Software Development with Formally Based Tools. Electronic Notes in Theoretical Computer Science, 2004, 108, 11-19.	0.9	9
17	On the Role of Formal Methods in Software Certification: An Experience Report. Electronic Notes in Theoretical Computer Science, 2009, 238, 3-9.	0.9	9
18	Model-based construction and verification of critical systems using composition and partial refinement. Formal Methods in System Design, 2010, 37, 265-294.	0.8	9

#	ARTICLE	IF	CITATIONS
19	A Formal Method for Developing Provably Correct Fault-Tolerant Systems Using Partial Refinement and Composition. Lecture Notes in Computer Science, 2009, , 173-189.	1.3	9
20	A Formal Statement of the MMS Security Model. , 1984, , .		8
21	A strategy for efficiently verifying requirements. Software Engineering Notes: an Informal Newsletter of the Special Interest Committee on Software Engineering / ACM, 2003, 28, 28-37.	0.7	8
22	Automatic Program Generation from Formal Specifications using APTS. , 2008, , 93-113.		8
23	Automatic generation of state invariants from requirements specifications. Software Engineering Notes: an Informal Newsletter of the Special Interest Committee on Software Engineering / ACM, 1998, 23, 56-69.	0.7	7
24	Direct generation of invariants for reactive models. , 2012, , .		4
25	High assurance human-centric decision systems. , 2013, , .		4
26	On Model-Based Software Development. , 2013, , 49-60.		4
27	Developing High Assurance Systems: On the Role of Software Tools. Lecture Notes in Computer Science, 2003, , 159-164.	1.3	3
28	Applying infinite state model checking and other analysis techniques to tabular requirements specifications of safety-critical systems. Design Automation for Embedded Systems, 2008, 12, 97-137.	1.0	3
29	Developing High Quality Software with Formal Methods: What Else Is Needed?. Lecture Notes in Computer Science, 2005, , 13-19.	1.3	1
30	Generating optimized code from SCR specifications. ACM SIGPLAN Notices, 2006, 41, 135-144.	0.2	0
31	RE Theory Meets Software Practice: Lessons from the Software Development Trenches. , 2007, , .		0
32	Property templates for checking source code security. , 2017, , .		0