JoÃ£o M LourenÃ§o

List of Publications by Year
in descending order

Source: https://exaly.com/author-pdf/6460947/publications.pdf

Version: 2024-02-01

		1937632	1720014
37	125	4	7
papers	citations	h-index	g-index
39	39	39	92
all docs	docs citations	times ranked	citing authors

#	Article	IF	Citations
1	Itâ€™s about Thyme: On the design and implementation of a time-aware reactive storage system for pervasive edge computing environments. Future Generation Computer Systems, 2021, 118, 14-36.	7.5	7
2	Discovering Concurrency Errors. Lecture Notes in Computer Science, 2018, , 34-60.	1.3	8
3	Verifying Real-World Software with Contracts for Concurrency. Lecture Notes in Computer Science, 2018, , 70-73.	1.3	1
4	Verifying Concurrent Programs Using Contracts. , 2017, , .		5
5	GOCRGO and GOGO. , 2017, , .		1
6	Towards a persistent publish/subscribe system for networks of mobile devices. , 2017, , .		2
7	Towards the Opportunistic Combination of Mobile Ad-hoc Networks with Infrastructure Access. , 2016, , .		4
8	Ephemeral Data Storage for Networks of Hand-Held Devices. , 2016, , .		5
9	Pot. Transactions on Architecture and Code Optimization, 2016, 13, 1-24.	2.0	3
10	A Suite of Java Message-passing Benchmarks to Support the Validation of Testing Models, Criteria and Tools. Procedia Computer Science, 2016, 80, 2226-2230.	2.0	2
11	A Hardware Approach to Detect, Expose and Tolerate High Level Data Races. , 2016, , .		1
12	Special issue on testing, analysis and debugging of concurrent programs. Software Testing Verification and Reliability, 2015, 25, 165-166.	2.0	0
13	Supporting Multiple Data Replication Models in Distributed Transactional Memory. , 2015, , .		4
14	Extracting static and dynamic structural information from java concurrent programs for coverage testing. , 2015, , .		5
15	Boosting locality in multi-version partial data replication. , 2015, , .		1
16	Dynamic Validation of Contracts in Concurrent Code. Lecture Notes in Computer Science, 2015, , 555-564.	1.3	3
17	Framework Support for the Efficient Implementation of Multi-version Algorithms. Lecture Notes in Computer Science, 2015, , 166-191.	1.3	0
18	On Monitoring C/C++ Transactional Memory Programs. Lecture Notes in Computer Science, 2014, , 73-87.	1.3	0

#	Article	IF	Citations
19	Efficient support for inâ€place metadata in Java software transactional memory. Concurrency Computation Practice and Experience, 2013, 25, 2394-2411.	2.2	4
20	Precise Detection of Atomicity Violations. Lecture Notes in Computer Science, 2013, , 8-23.	1.3	11
21	MacroDB: Scaling Database Engines on Multicores. Lecture Notes in Computer Science, 2013, , 607-619.	1.3	2
22	Software Component Replication for Improved Fault-Tolerance: Can Multicore Processors Make It Work?. Lecture Notes in Computer Science, 2013, , 173-180.	1.3	0
23	On the Relevance of Total-Order Broadcast Implementations in Replicated Software Transactional Memories. Lecture Notes in Computer Science, 2013, , 49-60.	1.3	1
24	Verification of Snapshot Isolation in Transactional Memory Java Programs. Lecture Notes in Computer Science, 2012, , 640-664.	1.3	5
25	Efficient Support for In-Place Metadata in Transactional Memory. Lecture Notes in Computer Science, 2012, , 589-600.	1.3	3
26	Practical verification of high-level dataraces in transactional memory programs. , 2011, , .		3
27	Detecting concurrency anomalies in transactional memory programs. Computer Science and Information Systems, 2011, 8, 534-548.	1.0	3
28	Understanding Transactional Memory (Extended Abstract). Lecture Notes in Computer Science, 2011, , 1-2.	1.3	0
29	Special Session on Debugging. Lecture Notes in Computer Science, 2011, , 24-28.	1.3	0
30	Open virtualization framework for testing ground systems. , 2010, , .		3
31	Detection of Transactional Memory anomalies using static analysis. , 2010, , .		8
32	Understanding the behavior of transactional memory applications. , 2009, , .		10
33	Unifying Memory and Database Transactions. Lecture Notes in Computer Science, 2009, , 349-360.	1.3	4
34	Developing libraries using software transactional memory. Computer Science and Information Systems, 2008, 5, 103-117.	1.0	2
35	Topic 1: Support Tools and Environments. Lecture Notes in Computer Science, 2008, , 1-2.	1.3	0
36	Testing patterns for software transactional memory engines. , 2007, , .		7

#	Article	IF	Citations
37	Fiddle: A Flexible Distributed Debugger Architecture. Lecture Notes in Computer Science, 2001, , 821-830.	1.3	4