Simon Peyton Jones

List of Publications by Year
in descending order

Source: https://exaly.com/author-pdf/10446287/publications.pdf

Version: 2024-02-01

81
papers

3,662
citations

687363

13
h-index

345221

36
g-index

82
all docs

82
docs citations

82
times ranked

761
citing authors

#	Article	IF	Citations
1	Build systems Ã la carte: Theory and practice. Journal of Functional Programming, 2020, 30, .	0.8	7
2	A quick look at impredicativity. , 2020, 4, 1-29.		10
3	Lower your guards: a compositional pattern-match coverage checker. , 2020, 4, 1-30.		6
4	Higher-order type-level programming in Haskell. , 2019, 3, 1-26.		6
5	Guarded impredicative polymorphism. , 2018, , .		12
6	Pattern synonyms. ACM SIGPLAN Notices, 2018, 51, 80-91.	0.2	2
7	Type variables in patterns. , 2018, , .		5
8	Guarded impredicative polymorphism. ACM SIGPLAN Notices, 2018, 53, 783-796.	0.2	1
9	Type variables in patterns. ACM SIGPLAN Notices, 2018, 53, 94-105.	0.2	2
10	Compiling without continuations. , 2017, , .		15
11	Levity polymorphism. , 2017, , .		5
12	Levity polymorphism. ACM SIGPLAN Notices, 2017, 52, 525-539.	0.2	0
13	Compiling without continuations. ACM SIGPLAN Notices, 2017, 52, 482-494.	0.2	3
14	Pattern synonyms. , 2016, , .		21
15	Safe zero-cost coercions for Haskell. Journal of Functional Programming, 2016, 26, .	0.8	12
16	A Reflection on Types. Lecture Notes in Computer Science, 2016, , 292-317.	1.3	10
17	Injective type families for Haskell. ACM SIGPLAN Notices, 2016, 50, 118-128.	0.2	3
18	Injective type families for Haskell. , 2015, , .		15

#	Article	IF	Citations
19	ICFP 2004. ACM SIGPLAN Notices, 2015, 50, 23-34.	0.2	0
20	Closed type families with overlapping equations. , 2014, , .		56
21	Modular, higher-order cardinality analysis in theory and practice. , 2014, , .		13
22	Modular, higher-order cardinality analysis in theory and practice. ACM SIGPLAN Notices, 2014, 49, 335-347.	0.2	2
23	Exploiting vector instructions with generalized stream fusion. , 2013, , .		20
24	Exploiting vector instructions with generalized stream fusion. ACM SIGPLAN Notices, 2013, 48, 37-48.	0.2	1
25	Equality proofs and deferred type errors. , 2012, , .		28
26	Giving Haskell a promotion. , 2012, , .		131
27	Equality proofs and deferred type errors. ACM SIGPLAN Notices, 2012, 47, 341-352.	0.2	7
28	<scp>OutsideIn(X)</scp>Modular type inference with local assumptions. Journal of Functional Programming, 2011, 21, 333-412.	0.8	96
29	Generative type abstraction and type-level computation. , 2011, , .		26
30	Generative type abstraction and type-level computation. ACM SIGPLAN Notices, 2011, 46, 227-240.	0.2	3
31	Multicore garbage collection with local heaps. ACM SIGPLAN Notices, 2011, 46, 21-32.	0.2	10
32	Let should not be generalized. , 2010, , .		23
33	Runtime support for multicore Haskell. , 2009, , .		67
34	Complete and decidable type inference for GADTs. , 2009, , .		68
35	Complete and decidable type inference for GADTs. ACM SIGPLAN Notices, 2009, 44, 341-352.	0.2	8
36	Type checking with open type functions. ACM SIGPLAN Notices, 2008, 43, 51-62.	0.2	15

#	Article	IF	Citations
37	Parallel generational-copying garbage collection with a block-structured heap. , 2008, , .		58
38	FPH. ACM SIGPLAN Notices, 2008, 43, 295-306.	0.2	4
39	Type checking with open type functions. , 2008, , .		88
40	Scrap Your Type Applications. Lecture Notes in Computer Science, 2008, , 2-27.	1.3	2
41	System F with type equality coercions. , 2007, , .		143
42	Data parallel Haskell. , 2007, , .		136
43	Call-pattern specialisation for Haskell programs. ACM SIGPLAN Notices, 2007, 42, 327-337.	0.2	8
44	Faster laziness using dynamic pointer tagging. ACM SIGPLAN Notices, 2007, 42, 277-288.	0.2	4
45	Practical type inference for arbitrary-rank types. Journal of Functional Programming, 2007, 17, 1-82.	0.8	123
46	Faster laziness using dynamic pointer tagging. , 2007, , .		21
47	A history of Haskell. , 2007, , .		120
48	Simple unification-based type inference for GADTs. , 2006, , .		185
49	Making a fast curry: push/enter vs. eval/apply for higher-order languages. Journal of Functional Programming, 2006, 16, 415-449.	0.8	31
50	Simple unification-based type inference for GADTs. ACM SIGPLAN Notices, 2006, 41, 50-61.	0.2	43
51	Boxy types. , 2006, , .		30
52	Associated type synonyms. ACM SIGPLAN Notices, 2005, 40, 241-253.	0.2	21
53	Associated type synonyms. , 2005, , .		133
54	Associated types with class. , 2005, , .		112

#	ARTICLE	IF	CITATIONS
55	Scrap your boilerplate with class. ACM SIGPLAN Notices, 2005, 40, 204-215.	0.2	12
56	Associated types with class. ACM SIGPLAN Notices, 2005, 40, 1-13.	0.2	13
57	Scrap your boilerplate with class. , 2005, , .		61
58	Extending the Haskell foreign function interface with concurrency. , 2004, , .		23
59	Scrap more boilerplate. ACM SIGPLAN Notices, 2004, 39, 244-255.	0.2	14
60	Making a fast curry. , 2004, , .		8
61	Constructed product result analysis for Haskell. Journal of Functional Programming, 2004, 14, 211-245.	0.8	7
62	Scrap more boilerplate. , 2004, , .		56
63	Scrap your boilerplate. , 2003, , .		174
64	Scrap your boilerplate. ACM SIGPLAN Notices, 2003, 38, 26-37.	0.2	65
65	Optimistic evaluation. ACM SIGPLAN Notices, 2003, 38, 287-298.	0.2	5
66	Template meta-programming for Haskell. , 2002, , .		308
67	Template meta-programming for Haskell. ACM SIGPLAN Notices, 2002, 37, 60-75.	0.2	130
68	Asynchronous exceptions in Haskell. ACM SIGPLAN Notices, 2001, 36, 274-285.	0.2	8
69	Porting the Clean Object I/O Library to Haskell. Lecture Notes in Computer Science, 2001, , 194-213.	1.3	6
70	Composing contracts. ACM SIGPLAN Notices, 2000, 35, 280-292.	0.2	37
71	A semantics for imprecise exceptions. , 1999, , .		53
72	Imprecise Exceptions, Co-Inductively. Electronic Notes in Theoretical Computer Science, 1999, 26, 122-141.	0.9	4

#	Article	IF	Citations
73	A semantics for imprecise exceptions. ACM SIGPLAN Notices, 1999, 34, 25-36.	0.2	11
74	Calling hell from heaven and heaven from hell. , 1999, , .		53
75	Let-floating. ACM SIGPLAN Notices, 1996, 31, 1-12.	0.2	5
76	Let-floating. , 1996, , .		57
77	Composing Haggis. Eurographics, 1995, , 85-101.	0.4	15
78	Programming Reactive Systems in Haskell. Workshops in Computing, 1995, , 50-65.	0.4	1
79	Report on the programming language Haskell. ACM SIGPLAN Notices, 1992, 27, 1-164.	0.2	539
80	A Parallel Functional Database on GRIP. Workshops in Computing, 1992, , 1-24.	0.4	1
81	Using Futurebus in a fifth-generation computer. Microprocessors and Microsystems, 1986, 10, 69-76.	2.8	6