| Record Nr. | UNISA996466428003316 |
|-------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Titolo | Functional and Logic Programming [[electronic resource]] : 14th International Symposium, FLOPS 2018, Nagoya, Japan, May 9–11, 2018, Proceedings / / edited by John P. Gallagher, Martin Sulzmann |
| Pubbl/distr/stampa | Cham : , : Springer International Publishing : , : Imprint : Springer, , 2018 |
| ISBN | 3-319-90686-0 |
| Edizione | [1st ed. 2018.] |
| Descrizione fisica | 1 online resource (XVI, 281 p. 74 illus.) |
| Collana | Theoretical Computer Science and General Issues, , 2512-2029 ; ; 10818 |
| Disciplina | 005.1 |
| Soggetti | Software engineering |
| | Compilers (Computer programs) |
| | Computer programming |
| | Computers |
| | Professions |
| | Electronic digital computers—Evaluation |
| | Computers, Special purpose |
| | Software Engineering |
| | Compilers and Interpreters |
| | Programming Techniques |
| | The Computing Profession |
| | System Performance and Evaluation |
| | Special Purpose and Application-Based Systems |
| Lingua di pubblicazione | Inglese |
| Formato | Materiale a stampa |
| Livello bibliografico | Monografia |
| Sommario/riassunto | This book constitutes the proceedings of the 14th International Symposium on Functional and Logic Programming, FLOPS 2018, held in Nagoya, Japan, in May 2018. The 17 papers presented in this volume were carefully reviewed and selected from 41 submissions. They cover all aspects of the design, semantics, theory, applications, implementations, and teaching of declarative programming focusing on |

1.

topics such as functional-logic programming, re-writing systems, formal methods and model checking, program transformations and program refinements, developing programs with the help of theorem provers or SAT/SMT solvers, language design, and implementation issues.