1. Record Nr. UNISA996465707603316

Titolo	Third International Conference on Logic Programming [[electronic resource]] : Imperial College of Science and Technology, London, United Kingdom, July 14-18, 1986. Proceedings / / edited by Ehud Shapiro
Pubbl/distr/stampa	Berlin, Heidelberg : , : Springer Berlin Heidelberg : , : Imprint : Springer, , 1986
ISBN	3-540-39831-7
Edizione	[1st ed. 1986.]
Descrizione fisica	1 online resource (IX, 723 p.)
Collana	Lecture Notes in Computer Science, , 0302-9743 ; ; 225
Disciplina	005.1015113
Soggetti	Computer logic Artificial intelligence Logics and Meanings of Programs Artificial Intelligence
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Bibliographic Level Mode of Issuance: Monograph
Nota di contenuto	The role of logic programming in the Fifth Generation Computer Project -- An abstract machine for restricted AND-parallel execution of logic programs -- Efficient management of backtracking in AND-Parallelism -- An intelligent backtracking algorithm for parallel execution of logic programs -- Delta Prolog: A distributed backtracking extension with events -- OLD resolution with tabulation -- Logic programs and alternation -- Intractable unifiability problems and backtracking -- On the complexity of unification sequences -- How to invent a Prolog machine -- A sequential implementation of Parlog -- A GHC abstract machine and instruction set -- A Prolog processor based on a pattern matching memory device -- An improved version of Shapiro's model inference system -- A framework for ICAI systems based on inductive inference and logic programming -- Rational debugging in logic programming -- Using definite clauses and integrity constraints as the basis for a theory formation approach to diagnostic reasoning -- Some issues and trends in the semantics of logic programming -- Parallel logic programming languages -- P-Prolog: A parallel logic language based on exclusive relation -- Making exhaustive search programs

deterministic -- Compiling OR-parallelism into AND-parallelism -- Shared memory execution of committed-choice languages -- Logic program semantics for programming with equations -- On the semantics of logic programming languages -- Towards a formal semantics for concurrent logic programming languages -- Design of a Prolog-based machine translation system -- Parallel logic programming for numeric applications -- Sequential and concurrent deterministic logic grammars -- A parallel parsing system for natural language analysis -- Equivalences of logic programs -- Qualified answers and their application to transformation -- Procedures in Horn-clause programming -- Higher-order logic programming -- Abstract interpretation of Prolog programs -- Verifleation of Prolog programs using an extension of execution -- Detection and optimization of functional computations in Prolog -- Control of logic program execution based on the functional relation -- Declarative graphics -- Test-pattern generation for VLSI circuits in a Prolog environment -- Using Prolog to represent and reason about protein structure -- A New approach for introducing Prolog to naive users -- Prolog programming environments: Architecture and implementation -- Design overview of the NAIL! System -- A superimposed codeword indexing scheme for very large Prolog databases -- Interfacing Prolog to a persistent data store -- A general model to implement DIF and FREEZE -- Cyclic tree traversal -- Completeness of the SLDNF-resolution for a class of logic programs -- Choices in, and limitations of, logic programming -- Negation and quantifiers in NU-Prolog -- Gracefully adding negation and disjunction to Prolog -- Memory performance of Lisp and Prolog programs -- The design and implementation of a high-speed incremental portable Prolog compiler -- Compiler optimizations for the WAM -- Fast decompilation of compiled Prolog clauses -- Logic continuations -- Cut & Paste — defining the impure primitives of Prolog -- Tokio: Logic programming language based on temporal logic and its compilation to Prolog -- The OR-forest description for the execution of logic programs.