Robinet

ESOP 86

European Symposium on Programming, Saarbrücken, Federal Republic of Germany, March 17-19, 1986. Proceedings

Compiler generation from relational semantics.- Correctness of code generation from a two-level meta-language.- A general approach to the optimization of function calls.- Lifetime analysis for procedure parameters.- Another implementation technique for applicative languages.- Graal : A Functional Programming system with uncurryfied combinators and its reduction machine.- A safe approach to parallel combinator reduction (extended abstract).- Reducing recursion to iteration by algebraic extension.- Design and implementation of a generic, logic and functional programming language.- Pragmatic aspects of two-level denotational meta-languages.- An experiment in practical semantics.- The choice of programming primitives for SETL-like programming languages.- Algebraic semantics of exception handling.- Data refinement refined resume.- Proving entailment between conceptual state specifications.- Automatic parallelization of fortran programs in the presence of procedure calls.- Parallelization by semantic detection of reductions.- Denotational semantics of nets with nondeterminism.- specification of a tool for viewing program text.- A classical implementation for Prolog-II.- Programs as collections of communicating Prolog units.- An and-parallel execution model of logic programs.- Compiling Extended Concurrent Prolog -single queue compilation-.- A very intelligent backtracking method for Logic Programs.- An application of abstract interpretation of logic programs: Occur check reduction.- Experiences with the RAP system - a specification interpreter combining term rewriting and resolution.- Rewriting with a nondeterministic choice operator : From algebra to proofs.

Compiler generation from relational semantics.- Correctness of code generation from a two-level meta-language.- A general approach to the optimization of function calls.- Lifetime analysis for procedure parameters.- Another implementation technique for applicative languages.- Graal : A Functional Programming system with uncurryfied combinators and its reduction machine.- A safe approach to parallel combinator reduction (extended abstract).- Reducing recursion to iteration by algebraic extension.- Design and implementation of a generic, logic and functional programming language.- Pragmatic aspects of two-level denotational meta-languages.- An experiment in practical semantics.- The choice of programming primitives for SETL-like programming languages.- Algebraic semantics of exception handling.- Data refinement refined resume.- Proving entailment between conceptual state specifications.- Automatic parallelization of fortran programs in the presence of procedure calls.- Parallelization by semantic detection of reductions.- Denotational semantics of nets with nondeterminism.- specification of a tool for viewing program text.- A classical implementation for Prolog-II.- Programs as collections of communicating Prolog units.- An and-parallel execution model of logic programs.- Compiling Extended Concurrent Prolog -single queue compilation-.- A very intelligent backtracking method for Logic Programs.- An application of abstract interpretation of logic programs: Occur check reduction.- Experiences with the RAP system - a specification interpreter combining term rewriting and resolution.- Rewriting with a nondeterministic choice operator : From algebra to proofs.



53,49 €
49,99 € (zzgl. MwSt.)

Lieferfrist: bis zu 10 Tage

Artikelnummer: 9783540164425
Medium: Buch
ISBN: 978-3-540-16442-5
Verlag: Springer
Erscheinungstermin: 01.03.1986
Sprache(n): Englisch
Auflage: 1. Auflage 1986
Serie: Lecture Notes in Computer Science
Produktform: Kartoniert
Gewicht: 1200 g
Seiten: 380
Format (B x H): 155 x 235 mm

Kundenservice Verlag Dr. Otto Schmidt KG
Neumannstraße 10, 40235 Düsseldorf | kundenservice@otto-schmidt.de | 0221-93738-997 (Inland)