TAPSOFT '87 Proceedings of the International Joint Conference on Theory and Practice of Software Development Pisa, Italy, March 1987.

Volume 1: Advanced Seminar on Foundations of Innovative Software Development I and Colloquium on Trees in Algebra and Programming (CAAP'87) (Lecture Notes in Computer Science Vol. 249) XIV+289 pages, Springer Verlag, Berlin—Heidelberg—New York—Tokio, 1987. Edited by Hartmut Ehrig, Robert Kowalski, Giorgio Levi and Ugo Montanari.

Volume 2: Advanced Seminar on Foundations of Innovative Software Development II and Colloquium on Functional and Logic Programming and Specifications (CFLP) (Lecture Notes in Computer Science Vol. 250) XIV+336 pages, Springer Verlag, Berlin—Heidelberg—New York—Tokio, 1987. Edited by Hartmut Ehrig, Robert Kowalski, Giorgio Levi and Ugo Montanari.

These two books contain a selected collection of papers presented at TAPSOFT '87 held in Pisa, Italy, March 1987.

TAPSOFT '87 consists of three parts:

i, Advanced Seminar on Foundations of Innovative Software Development concerns new directions in software development on the basis of recent technological and theoretical advances.

ii, Colloquium on Trees in Algebra and Programming covers the formal aspect and properties of trees, and more generally, combinatorial and algebraic structures in all fields of Computer Science. Besides the customary topics, CAAP includes contributions related to specifications, communicating systems and type theory.

iii, Colloquium on Functional and Logic Programming and Specifications focuses on those aspects of Functional and Logic Programming which are most important in innovative software development.

Contents of Volume 1

Contents of Volume 2

J. A. Goguen and J. Meseguer: Models and Equality for Logical Programming, K. Furukawa: Fifth Generation Computer Project: Current Research Activity and Future Plans, A. Piperno: A Compositive Abstraction Algorithm for Combinatory logic, J. Y. Girard and Y. Lafont: Linear Logic and Lazy Computation, D. Clément: The Natural Dynamic Semantics of Mini-Standard ML, Z. Farkas: Listlog — a Prolog Extension for List Processing, R. Barbuti, P. Mancarella, D. Pedreschi and F. Turini: Intensional Negation of Logic Programs: Examples and Implementation Techniques, P. Van Roy, B. Demoen and Y. D. Willems: Improving the Execution Speed of Compiled Prolog with Modes, Clause Selection, and Determinism, C. Percebois, I. Futó, I. Durand, C. Simon and B. Bonhoure: Simulation Results of a Multiprocessor Prolog Architecture Based on a Distributed AND/OR Graph, G. Lindstrom, L. George and D. Yeh: Generating Efficient Code from Strictness Annotations, S. Finn: Hoisting: Lazy Evaluation in a Cold Climate, W. Drabent and J. Maluszynski: Inductive Assertion Method for Logic Programs, A. Pettorossi and A. Skowron: Higher Order Generalization in Program Derivation, M. Thomas: Implementing Algebraically Specified Abstract Data Types in an Imperative Programming Language, K. L. Clark and I. T. Foster: A Declarative Environment for Concurrent Logic Programming, D. H. D. Warren: Or-Parallel Execution Models of Prolog, M. Bellia: Retractions: a Functional Paradigm for Logic Programming, P. G. Bosco, E. Giovannetti and C. Moiso: Refined Strategies for Semantic Unification, V. Breazu-Tannen and T. Coquand: Extensional Models for Polymorphism, R. Harper, R. Milner and M. Tofte: A Type Discipline for Program Modules, C. Beierle and A. Voss: Theory and Practice of Canonical Term Functors in Abstract Data Type Specifications.

These well edited interesting volumes present the state of the art in theory and practice of software development. It is recommended for those people interested in the latest results of the field.

S. Vágvölgyi