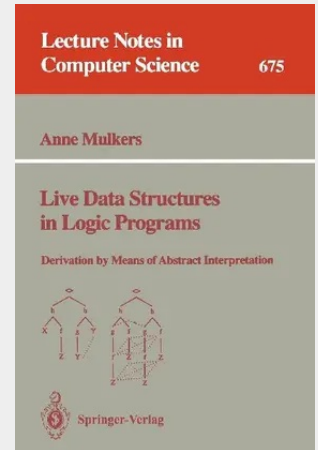


Live Data Structures in Logic Programs

Derivation by Means of Abstract Interpretation

A common hazard for implementations of applicative programming languages is the excessive creation of garbage cells during program execution. The available run-time garbage collecting processes are expensive in time and often require a periodic disruption of the program execution. The present book addresses the problem of memory re-use for logic programs through program analysis rather than by run-time garbage collection. The static analysis is constructed as an application of abstract interpretation for logic programs. The starting point is a previously developed application of integrated type and mode analysis, which basically supplies a description of the logical terms to which program variables can be bound at run time. The contribution of the book consists of a modular extension of the abstract domain and operations in order to derive run-time properties concerning the sharing and liveness of term substructures dynamically created during program execution. Also illustrated is how such information can be used by the compiler to improve the allocation of run-time structures and to automatically introduce destructive assignments in logic languages in a safe and transparent way, thereby shifting some part of the run-time storage reclamation overhead to compile time.

A common hazard for implementations of applicative programming languages is the excessive creation of garbage cells during program execution. The available run-time garbage collecting processes are expensive in time and often require a periodic disruption of the program execution. The present book addresses the problem of memory re-use for logic programs through program analysis rather than by run-time garbage collection. The static analysis is constructed as an application of abstract interpretation for logic programs. The starting point is a previously developed application of integrated type and mode analysis, which basically supplies a description of the logical terms to which program variables can be bound at run time. The contribution of the book consists of a modular extension of the abstract domain and operations in order to derive run-time properties concerning the sharing and liveness of term substructures dynamically created during program execution. Also illustrated is how such information can be used by the compiler to improve the allocation of run-time structures and to automatically introduce destructive assignments in logic languages in a safe and transparent way, thereby shifting some part of the run-time storage reclamation overhead to compile time.



53,49 €

49,99 € (zzgl. MwSt.)

Lieferfrist: bis zu 10 Tage

Artikelnummer: 9783540566946

Medium: Buch

ISBN: 978-3-540-56694-6

Verlag: Springer

Erscheinungstermin: 27.05.1993

Sprache(n): Englisch

Auflage: Erscheinungsjahr 1993

Serie: Lecture Notes in Computer Science

Produktform: Kartoniert

Gewicht: 365 g

Seiten: 227

Format (B x H): 155 x 235 mm

