INFORMATICA, 1999, Vol. 10, No. 1, 3–4

Guest Editorial: Special Issue on Programming Theory, Information System Engineering, Software Engineering, and Artificial Intelligence

Albertas ČAPLINSKAS
Institute of Mathematics and Informatics
Akademijos 4, 2600 Vilnius, Lithuania
e-mail: alcapl@ktl.mii.lt

The Special Issue contains seven papers, covering significant aspects of programming theory, multimedia information systems, software engineering and AI.

Three papers are devoted to programming theory. The paper of Tarmo Uustalu and Varmo Vene is an intermediate report of an ongoing research project on program construction for typed functional languages and follows the tradition of categorical approach inspired by R. S. Bird. It discusses the dual construction of apomorphisms (morphisms similar to catamorphisms) and introduces a new construction called paramorphism. Apomorphisms are a generalization of anamorphisms. The calculation properties of paramorphisms are similar to those of catamorphisms. Every paramorphism is the composition of the left projection and a catamorphism. On the other hand, paramorphisms can be viewed as a generalization of catamorphisms. The function definition schemes of iteration and coiteration can be modelled by catamorphisms and anamorphisms, primitive recursion can be captured by paramorphisms and any function that can be characterized similarly to the factorial function definable as a paramorphism. The paper shows by examples that primitive corecursion is a useful function definition scheme. The primitive recursion is a generalization of iteration. The value of an iterative function for a given argument depends solely on the values for immediate subparts, the value of a primitive recursive function may additionally depend on immediate subparts directly. In order to generalise the coiteration, the paper introduces two new constructions, histomorphisms, and futumorphisms. Similarly to paramorphisms, histomorphisms are a generalization of catamorphisms. Histomorphisms capture course-of-value iteration (e.g., natural definition scheme of the Fibonacci function and closely similar functions). Futumorphisms are dual to histomorphisms. The argument of a coiterative function for a value may only determine the argument for the immediate subparts of the value, the value of a course-of-value iterative function depends on the values for any subparts of the argument. Primitive (co) recursion and course-of-value (co)iteration facilitate to write programs and to prove properties about programs, because they are more “liberal” as (co)iteration.

Aleksey L. Gomozov and Larisa I. Stanevichiene aim at expressing adequately the common essence of formal descriptions of the context-free languages by D-graphs, pushdown automata and context-free grammars. They define context-free expressions that generalise the regular one. The subclass of context-free expressions describes the sets that can be defined by context-free grammars without self-embedding symbols. It means that every context-free expression defines a context-free language. The subclass of context-free expressions can be split into coiterating and pseudo-coiterating expressions. Pseudo-coiterating context-free expressions characterise the regular sets. The paper proposes an algorithm to decide whether a context-free expression is pseudo-coiterating or not. Consequently, the paper defines the sufficient conditions of regularity of context free-language, because the pseudo-coiterating pushdown automata characterise the regular sets.

Vytautas Čyras investigates how to use the structural blanks approach in the consistency proof of data dependencies in loop programs and uses the parallel algorithm for solving triangular systems of equations proposed by J.A. Yang and Y. Choo as a sample. The structural blanks approach aims at expressing solutions to mutually dependent recurrence in the form of reusable program components defining loops over arrays. It distinguishes between functional and structural components (modules). The paper demonstrates how to represent the essential part of the Yang and Choo algorithm in the form of sequential and parallel structural components and presents a detailed consistency proof.

Algimantas Juozapavičius and Richard E. Blake present comparative analysis of recent trends in the area of multidimensional, spatial, temporal indices and data structures. They also discuss multidimensional access methods, search and reasoning using indices and other implementation issues.

Three last papers are written by young researchers. Janis Plume proposes an instrumental approach to data conversion. He examines what kinds of activities are needed to develop data conversion software, how those activities can be integrated into the life cycle model, and proposes to support the software process by a set of loosely integrated small tools. The advantage of the proposed approach is that each tool can be improved or replaced without affecting other tools.

Kristina Lapin examines the architectural and implementation issues of a software system that is intended to support the preparation of technical drawings for electroplating lines. She supposes that the drawing preparation process can be regarded as a combination of two tasks: a configuration of the electroplating line and the final document creation. The paper suggests using the component-oriented approach to cope with the first task and the ISO 8879 based generic coding approach to cope with the second one.

Tomasz Lukaszewski proposes an indices-based algorithm to update the evidence in the Dempster-Shafer theory. The proposed algorithm allows us to remove and change some pieces of evidence in a more effective way than the classical algorithms based on the idea of carrying out all combinations again except the removed or changed ones.

I wish to express my appreciation to the authors, reviewers, and all the people involved in the preparation of this Special Issue. I believe that each of the papers contributes to information and computer sciences and I hope that the ideas proposed or extended by the authors will be of interest to the readers of INFORMATICA.