

Contents

High Level Specification of I/O in Functional Languages <i>P. Achten, J. van Groningen and R. Plasmeijer</i>	1
GRIT: Guy's RISC Implementation of the Three Instruction Machine <i>G. Argo</i>	18
A Logical Framework for Program Analysis <i>G. Burn</i>	30
Analysing Functions by Projection-Based Backward Abstraction <i>K. Davis</i>	43
Abstract Interpretation of Higher Order Functions Using Concrete Data Structures (Summary) <i>A.B. Ferguson and J. Hughes</i>	57
The Glasgow Haskell Compiler: A Retrospective <i>C. Hall, K. Hammond, W. Partain, S.L. Peyton Jones and P. Wadler</i>	62
Improving Persistent Data Manipulation for Functional Languages <i>K. Hammond, D. McNally, P.M. Sansom and P. Trinder</i>	72
A Loop-Detecting Interpreter for Lazy, Higher-Order Programs <i>J. Hughes and A.B. Ferguson</i>	85
A Semantics for Locally Bottom-Avoiding Choice <i>J. Hughes and A. Moran</i>	102
A Certain Loss of Identity <i>G. Jones and M. Sheeran</i>	113
Programming with Constructor Classes (Preliminary Summary) <i>M.P. Jones</i>	122
Combining Monads <i>D.J. King and P. Wadler</i>	134

Avoiding Unnecessary Updates <i>J. Launchbury, A. Gill, J. Hughes, S. Marlow, S. Peyton Jones and P. Wadler</i>	144
Deforestation for Higher-Order Functions <i>S. Marlow and P. Wadler</i>	154
Hazard Algebra and the Design of Asynchronous Automata <i>E. Meijer</i>	166
Generating Netlists from Executable Circuit Specifications in a Pure Functional Language <i>J.T. O'Donnell</i>	178
The nofib Benchmark Suite of Haskell Programs <i>W. Partain</i>	195
Heap Profiling of a Lazy Functional Compiler <i>C. Runciman and D. Wakeling</i>	203
LZW Text Compression in Haskell <i>P. Sanders and C. Runciman</i>	215
Profiling Lazy Functional Programs <i>P.M. Sansom and S.L. Peyton Jones</i>	227
On Program Transformation in the Glasgow Haskell Compiler <i>A. Santos and S. L. Peyton Jones</i>	240
Graphical User Interfaces for Haskell <i>D.C. Sinclair</i>	252
Formulating Haskell <i>S. Thompson</i>	258
Author Index	269