Seminal Research Articles in Programming Languages

Abstract

The research articles listed here cover the period 1963 to the present. They are all important articles, and they are relevant to this offering of the course.

1 List of Articles

Here is a list of research articles that are relevant to the topics covered in CS6110 Advanced Programming Languages for the Spring 2015 offering. One of the assignments in the course will be to read one of these articles, summarize its key points, relate them to the course, and suggest interesting research questions related to the article.

It is also possible for students to suggest articles that they believe are important and relevant. We will consider adding them if appropriate.

1. A Basis for a Mathematical Theory of Computation [15].
2. An axiomatic basis for computer programming [10].
3. A critique of the foundations of Hoare style programming logics [18].
4. Notes on data structuring [11].
5. Towards a theory of type structure [22].
6. Data types as lattices [24].
7. Call-by-name, call-by-value, and the λ-calculus [19].
8. Equality in Lazy Computation Systems [12].
9. A Structural Approach to Operational Semantics [21].
10. Natural Semantics [13].
11. LCF considered as a programming language [20].
12. A theory of type polymorphism in programming [16].
13. The Essence of Algol [23].
14. The Essence of ML [17].
15. Logical relations and the typed lambda calculus [25].
16. Monads for functional programming [26].
17. Proofs as Programs [5].
18. Intuitionistic type theory [14].
19. A Non-Type-Theoretic Definition of Martin-Löf's Types [1].
20. Partial Objects in Constructive Type Theory [7].
21. Constructing Type systems over an operational semantics [9].
22. An Abstract Semantics for Atoms in Nuprl [2].
23. Unguessable Atoms: A Logical Foundation for Security [6].
24. Towards a formally verified proof assistant [4].
25. A type theory with partial equivalence relations as types [3].

References

[1] Stuart F. Allen. A Non-type-theoretic Definition of Martin-Löf's Types. In Gries [8], pages 215–224.

[2] Stuart F. Allen. An Abstract Semantics for Atoms in Nuprl. Technical report, 2006.

[3] Abhishek Anand, Mark Bickford, Robert Constable, and Vincent Rahli. A type theory with partial equivalence relations as types. In *TYPES 2014*. 2014.

[4] Abhishek Anand and Vincent Rahli. Towards a formally verified proof assistant. In *International Conference on Interactive Theorem Proving*, pages 95–197, 2014.

[5] J. L. Bates and Robert L. Constable. Proofs as programs. *ACM Transactions of Programming Language Systems*, 7(1):53–71, 1985.

[6] Mark Bickford. Unguessable atoms: A logical foundation for security. In *Verified Software: Theories, Tools, Experiments, Second International Conference*, pages 30–53, Toronto, Canada, 2008. VSTTE 2008.

[7] Robert L. Constable and Scott Fraser Smith. Partial objects in constructive type theory. In Gries [8], pages 183–193.

[8] D. Gries, editor. *Proceedings of the 2^{nd} IEEE Symposium on Logic in Computer Science*. IEEE Computer Society Press, June 1987.

[9] Robert Harper. Constructing type systems over an operational semantics. *J. Symbolic Computing*, 14(1):71–84, 1992.

[10] C. A. R. Hoare. An axiomatic basis for computer programming. *Communications of the ACM*, 12(10):576–580,583, 1969.

[11] C. A. R. Hoare. Notes on data structuring. In *Structured Programming*. Academic Press, New York, 1972.

[12] Douglas J. Howe. Equality in lazy computation systems. In *Proceedings of the 4^{th} IEEE Symposium on Logic in Computer Science*, pages 198–203, Asilomar Conference Center, Pacific Grove, California, June 1989. IEEE Computer Society Press.

[13] G. Kahn. Natural semantics. In G. Vidal-Naquet F. Brandenburg and M. Wirsing, editors, *STACS '87*, volume 247 of *Lecture Notes in Computer Science*, pages 22–39. Springer-Verlag, Berlin, 1987.

[14] Per Martin-Löf. *Intuitionistic Type Theory.* Number 1 in Studies in Proof Theory, Lecture Notes. Bibliopolis, Napoli, 1984.

[15] J. McCarthy. A basis for a mathematical theory of computation. In P. Braffort and D. Hirschberg, editors, *Computer Programming and Formal Systems*, pages 33–70. North-Holland, Amsterdam, 1963.

[16] Robin Milner. A theory of type polymorphism in programming. *Journal of computer and system sciences*, 17:363–371, 1978.

[17] John C. Mitchell and Robert Harper. The essence of ml. In *Proceedings of the 15th ACM Symposium on Principles of Programming Languages*, pages 28–46. ACM, 1988.

[18] M.J. O'Donnell. A critique of the foundations of hoare style programming logics. *Commications of the Association of Computing Machinery*, 25(12):927–35, 1982.

[19] Gordon Plotkin. Call-by-name, call-by-value, and the λ-calculus. *Journal of Theoretical Computer Science*, pages 125–59, 1975.

[20] Gordon D. Plotkin. LCF considered as a programming language. *Journal of Theoretical Computer Science*, 5:223–255, 1977.

[21] Gordon D. Plotkin. A structural approach to operational semantics. Technical Report DAIMI-FN-19, Aarhus University, Aarhus University, Computer Science Department, Denmark, 1981.

[22] John C. Reynolds. Towards a theory of type structure. In *Proceedings Colloque sur, la Programmation*, volume 19 of *Lecture Notes in Computer Science*, pages 408–23. Springer-Verlag, New York, 1974.

[23] John C. Reynolds. The essence of Algol. In J. de Bakker and J. van Vliet, editors, *International Symposium on Algorithmic Languages*, pages 345–372. North-Holland, Amsterdam, 1981.

[24] D. Scott. Data types as lattices. *SIAM J. Comput.*, 5:522–87, 1976.

[25] R. Statman. Logical relations and the typed lambda calculus. *Information and Control*, 65:85–97, 1985.

[26] Philip Wadler. Monads for functional programming. In *Advanced Functional Programming*, volume 925 of *Lecture Notes in Computer Science*. Springer, 1995.