

Computer Sciences Department
University of Wisconsin
1210 West Dayton Street
Madison, Wisconsin 53706

BIBLIOGRAPHY ON PROVING THE CORRECTNESS
OF COMPUTER PROGRAMS -- ADDITION NO. 1

by

Ralph L. London

Technical Report #104

December 1970

PREFACE

The continued research activity in proving the correctness of computer programs and the widespread interest in my previous program proving bibliography (London 1970a) encourage me to compile these additional updating citations. The selection criteria remain essentially unchanged; only the form of the citation is trivially changed. As before, I would be most interested in learning of corrections and additional citations to update matters.

This work is supported by the National Science Foundation under Grant GJ-583.



BIBLIOGRAPHY ON PROVING THE CORRECTNESS
OF COMPUTER PROGRAMS--ADDITION NO. 1

by

Ralph L. London

- Abdali, S. K. , 1970. On proving sequential machine designs.
- Aho, A. V. , Sethi, R. , and Ullman, J. D. , 1970. A formal approach to code optimization, Proc. of a Symposium on Compiler Optimization, SIGPLAN Notices, 5, 7, July, pp. 86-100.
- Aho, A. V. and Ullman, J. D. , 1970. Transformations on straight line programs - preliminary version, Second Annual ACM Symposium on Theory of Computing, ACM, New York, pp. 136-148.
- Ashcroft, E. A. , 1970. Mathematical logic applied to the semantics of computer programs, Ph.D. thesis, Imperial College.
- Ashcroft, E. A. and Manna, Z. , 1970a. Formalization of properties of parallel programs, Artificial Intelligence Memo AIM-110, Stanford University, February.
- Ashcroft, E. A. and Manna, Z. , 1970b., The translation of "go-to" programs to "while" programs, Computer Science Technical Report No. CS 188, Stanford University.
- Basford, R. L. , 1967. A matrix approach to program description and optimization, Ph.D. thesis, Columbia University.
- Batcher, K. E. , 1968. Sorting networks and their applications, Proc. 1968 Spring Joint Computer Conference, Vol. 32, Thompson Book Co. , Washington, pp. 307-314. See especially Appendices A and B, p. 313.
- Bekić, H. and Walk, K. , 1970. Formalization of storage properties, In Engeler, E. (ed.), 1971. Also IBM Laboratory Report LR 25.5.034, Vienna, July.
- Bell, J. R. , 1969. A new method for determining linear precedence functions for precedence grammars, Comm. ACM, 12, 10, October, pp. 567-569.

- Boyle, J. M. and Grau, A. A., 1970. An algorithmic semantics for Algol 60 identifier denotation, J. ACM, 17, 2, April, pp. 361-382.
- Bredt, T. H., 1970. The mutual exclusion problem, Computer Science Technical Report No. CS 173, Stanford University.
- Cooper, D. C., 1970. Programs for mechanical program verification, Machine Intelligence 6, (to appear). Also Computer Science Memorandum No. 13, University College of Swansea, July.
- de Bakker, J. W., 1970. A property of linear conditionals, In Engeler, E. (ed.), 1971.
- de Bakker, J. W. and Scott, D., 1969. A theory of programs.
- Dijkstra, E. W., 1970. Notes on structured programming, T. H. - Report 70-WSK-03, Technological University Eindhoven, The Netherlands, Second Edition, April.
- Elspas, B., Green, M. W., and Levitt, K. N., 1971. Software reliability, Computer, 4, 1, January/February, (to appear). See especially section IV.
- Elspas, B., Levitt, K. N., Waksman, A., and Waldinger, R. J., 1971. An assessment of techniques for proving program correctness, Project 8398 Final Report, Stanford Research Institute, (to appear).
- Engeler, E. (ed.), 1971. Semantics of Algorithmic Languages, Lecture Notes in Mathematics, vol. 188, Springer Verlag, Berlin-Heidelberg-New York, (to appear). This book contains numerous relevant papers to program proving; for lack of information not all such papers could be cited separately in this bibliography.
- Feldman, J. A., 1970. Toward automatic programming, Software Engineering Techniques, Buxton, J. N. and Randell, B. (eds.), Scientific Affairs Division, NATO, Brussels, Belgium.
- Foley, M., 1969. Proof of the recursive procedure Quicksort: A comparison of two methods, Master's dissertation, Queen's University of Belfast.
- Good, D. I., 1970. Toward a man-machine system for proving program correctness, Ph.D. thesis, University of Wisconsin.

- Green, C., 1969. Theorem proving by resolution as a basis for question-answering systems, Machine Intelligence 4, Meltzer, B. and Michie, D. (eds.), Edinburgh University Press, Edinburgh, pp. 183-205.
- Gruenberger, F., 1968. Program testing and validating, Datamation, 14, 7, July, pp. 39-47.
- Hansen, W. J., 1969. The impact of storage management on plex processing language implementation, Computer Science Technical Report No. CS 113, Stanford University, July. See especially Appendix L, Demonstration of the correctness of the Swym garbage collection algorithm, pp. 196-224.
- Henhapl, W., 1969. A storage model derived from axioms, IBM Laboratory Technical Report TR 25.100, Vienna.
- Henhapl, W. and Jones, C. B., 1970. The block concept and some possible implementations, with proofs of equivalence, IBM Laboratory Technical Report TR 25.104, Vienna.
- Hoare, C. A. R. Proof of a program: Find, Comm. ACM, (to appear).
- Holt, R. On the correctness of a state of a scheduler (approximate subject), Ph.D. thesis to be submitted to Cornell University.
- Holt, R., 1970. Comments on prevention of system deadlocks, Computer Science Technical Report No. 70-50, Cornell University, January.
- Hunt, B. R., 1970. A comment on axiomatic approaches to programming, Comm. ACM, 13, 7, July, p. 452.
- Igarashi, S., 1970. Semantics of Algol-like statements, In Engeler, E. (ed.), 1971. Also Artificial Intelligence Memo AIM-129, Stanford University, June.
- Jones, C. B. and Lucas, P., 1970. Proving correctness of implementation techniques, In Engeler, E. (ed.), 1971. Also IBM Laboratory Technical Report TR 25.110, Vienna, August.
- Kaplan, D. M., 1970. Proving things about programs, Presented at Fourth Annual Princeton Conference on Information Sciences and Systems.

- King, J. C. and Floyd, R. W., 1970. An interpretation oriented theorem prover over integers. Second Annual ACM Symposium on Theory of Computing, ACM, New York, pp. 169-179.
- Knuth, D. E. and Floyd, R. W., 1970. Notes on avoiding "go to" statements, Computer Science Technical Report No. CS 148, Stanford University, January.
- Lauer, P. E., 1970. Survey paper on program proving, Part 1, The constructive approach, March; Part 2, A functional approach, May; Part 3, An axiomatic approach based on a precondition function, May; Part 4, A relational approach, May; Research Memoranda 1-3, 5, Department of Computer Science, Queen's University of Belfast.
- Leavenworth, B., 1970. See Naur 1969.
- London, R. L., 1970a. Bibliography on proving the correctness of computer programs, Machine Intelligence 5, Meltzer, B. and Michie, D. (eds.), Edinburgh University Press, Edinburgh, pp. 569-580. Also Computer Sciences Technical Report No. 64, University of Wisconsin, Revised Printing, November 1970.
- London, R. L., 1970b. Proof of algorithms: A new kind of certification (Certification of Algorithm 245 TREESORT 3), Comm. ACM, 13, 6, June, pp. 371-373. See also note by K. A. Redish and reply by R. L. London in Short Communications Section, Comm. ACM, (to appear).
- London, R. L., 1970c. Proving programs correct: Some techniques and examples, BIT, 10, 2, pp. 168-182.
- London, R. L., 1970d. Experience with inductive assertions for proving programs correct, In Engeler, E. (ed.), 1971, pp. 236-251. Also Computer Sciences Technical Report No. 92, University of Wisconsin, May.
- London, R. L., 1970e. A correctness proof of the Fisher-Galler algorithm using inductive assertions, Computer Sciences Technical Report No. 102, University of Wisconsin, October.

- London, R. L., 1971. Software reliability through proving programs correct. To be presented at the International Symposium on Fault-Tolerant Computing, March. Abstract in Computer, 4, 1, January/February, (to appear).
- Lowe, T. C., 1969. Analysis of Boolean program models for time-shared, paged environments, Comm. ACM, 12, 4, April, pp. 199-205.
- Lucas, P., Lauer, P., and Stigleitner, H., 1968. Method and notation for the formal definition of programming languages, IBM Laboratory Technical Report TR 25.087, Vienna, June.
- Lucas, P. and Walk, K., 1969. On the formal description of PL/I, Annual Review in Automatic Programming, Halpern, M. I. and Shaw, C. J. (eds.), Vol. 6, Part 3, Pergamon Press.
- McGowan, C. L. Correctness results for lambda calculus implementations, Ph.D. thesis, Cornell University.
- McGowan, C. L., 1969. On the equivalence of mechanical evaluation strategies, Presented at SIGPLAN Symposium on Programming Language Definition. Also Computer Science Technical Report No. 69-37, Cornell University, July.
- McGowan, C. L., 1970a. The correctness of a modified SECD machine, Second Annual ACM Symposium on Theory of Computing, ACM, New York, pp. 149-157.
- McGowan, C. L., 1970b. Proving equivalence of implementations of the lambda calculus, In Rustin, R. (ed.), 1970.
- Malinovskiy, V. I., 1967. Equivalence transformations of characteristic complexes, I and II, Automatic Programming, Seminar No. 2 (Kiev), pp. 3-17 and 18-27, (Russian).
- Manna, Z., 1970a. Termination of programs represented as interpreted graphs, Proc. 1970 Spring Joint Computer Conference, Vol. 36, AFIPS Press, Montvale, N. J., pp. 83-89.
- Manna, Z., 1970b. Second-order mathematical theory of computation, Second Annual ACM Symposium on Theory of Computing, ACM, New York, pp. 158-168. Also Artificial Intelligence Memo AIM-111, Stanford University, March.

- Manna, Z., 1970c. The correctness of non-deterministic programs, Artificial Intelligence, 1, 1/2, Spring, pp. 1-26. Also Artificial Intelligence Memo No. 95, Stanford University, 1969.
- Manna, Z., 1970d. Termination of interpreted graphs.
- Manna, Z., 1970e. Mathematical theory of partial correctness, Computer Science Technical Report No. CS 189, Stanford University.
- Manna, Z. and McCarthy, J., 1970. Properties of programs and partial function logic, Machine Intelligence 5, Meltzer, B. and Michie, D. (eds.), Edinburgh University Press, Edinburgh, pp. 27-37.
- Manna, Z. and Ness, S., 1970. On the termination of Markov algorithms, Proc. of Third Hawaii International Conference on System Sciences, Western Periodicals Company, pp. 789-792 (in Part 2).
- Manna, Z. and Pnueli, A., 1970. Formalization of properties of functional programs, J. ACM, 17, 3, July, pp. 555-569.
- Manna, Z. and Waldinger, R. J., 1970. Towards automatic program synthesis. Comm. ACM, (to appear). Also in Engeler, E. (ed.), 1971; Artificial Intelligence Memo AIM-127, Stanford University, July; and Artificial Intelligence Group Technical Note 34, Stanford Research Institute, July.
- Maurer, W. D., 1970. Algorithm verification.
- Mills, H. D., 1970. Syntax-directed documentation for PL360, Comm. ACM, 13, 4, April, pp. 216-222.
- Milner, R., 1969. The difficulty of verifying a program with unnatural data representation, Computer Science Memorandum No. 3, University College of Swansea, January.
- Milner, R., 1970a. Program schemes and recursive function theory, Machine Intelligence 5, Meltzer, B. and Michie, D. (eds.), Edinburgh University Press, Edinburgh, pp. 39-58.
- Milner, R., 1970b. Algebraic theory of computable polyadic functions, Computer Science Memorandum No. 12, University College of Swansea, April.
- Milner, R., 1970c. A formal notion of simulation between programs, Computer Science Memorandum No. 14, University College of Swansea, October.

- Morris, J. H. Jr., 1970a. Another recursion induction principle, Computer Center Technical Report No. 38, University of California, Berkeley.
- Morris, J. H. Jr., 1970b. A correctness proof using recursively defined functions, Computer Center Technical Report No. 39, University of California, Berkeley. Also in Rustin, R. (ed.), 1970.
- Munteanu, E., 1967. Logical analysis of algorithms, II, Studii se cercetari Mat. Acad. RSR, 19, 3, pp. 415-422, (Rumanian).
- Naur, P., 1969. Programming by action clusters, BIT, 9, 3, pp. 250-258. See also review by Leavenworth, B. in Computing Reviews, 11, 7, July, 1970, pp. 396-397.
- Noonan, R. E., 1971. The "algebracizing" of programming, Ph.D. thesis to be submitted to Purdue University.
- Orgass, R. J., 1970. Some results concerning proofs of statements about programs, J. Computer and System Sciences, 4, 1, February, pp. 74-88.
- Painter, J. A., 1970. Effectiveness of an optimizing compiler for arithmetic expressions, Proc. of a Symposium on Compiler Optimization, SIGPLAN Notices, 5, 7, July, pp. 101-126.
- Park, D., 1970. Fixpoint induction and proofs of program semantics, Machine Intelligence 5, Meltzer, B. and Michie, D. (eds.), Edinburgh University Press, Edinburgh, pp. 59-78.
- Redish, K. A. See London 1970b.
- Rosen, B. K., 1970. Tree-manipulating systems and Church - Rosser theorems, Second Annual ACM Symposium on Theory of Computing, ACM, New York, pp. 117-127. See especially p. 123, col. 1.
- Rosenberg, A. L., 1970. Data graphs and addressing schemes, IBM Report RC-2747, January. Extended abstract in Second Annual ACM Symposium on Theory of Computing, ACM, New York, pp. 48-61.

- Rustin, R. (ed.), 1970. Formal Semantics of Programming Languages, Proc. of Symposium held at Courant Institute, September, (to appear).
- Scott, D., 1970. The lattice of flow diagrams.
- Sethi, R. and Ullman, J. D., 1970. The generation of optimal code for arithmetic expressions, J. ACM, 17, 4, October, pp. 715-728.
- Stark, R. H., 1969. A language for algorithms.
- Strong, H. R. Jr., 1970. Translating recursion equations into flow charts, Extended abstract, Second Annual ACM Symposium on Theory of Computing, ACM, New York, pp. 184-197.
- Tiernan, J. C., 1970. An efficient search algorithm to find the elementary circuits of a graph, Comm. ACM, 13, 12, December, pp. 722-726.
- Tonoyan, R. N., 1966. On the formal passage from the scheme of an algorithm to the computation scheme and the program scheme, Transactions of the Computing Center of the Academy of Science of the Armenian SSR in Erevansk, No. 4, pp. 78-103, (Russian).
- Wegner, P., 1970. Information structure models of programming languages, In Rustin, R., (ed.), 1970.
- Weiner, M., 1969. The degree of unsolvability of the problem of equivalence of two program schemes, Computer Science Memorandum No. 9, University College of Swansea.
- Williams, C. M., 1967. On the equivalence and optimization of sequential algorithmic processes, Ph.D. thesis, University of Texas. Also Computation Center Report No. TNN-68, June.
- Yohe, J. M., 1970. Best possible floating point arithmetic, Mathematics Research Center Technical Summary Report No. 1054, University of Wisconsin, March.
- Zimmermann, K., 1970. Loop optimization: Representation and proof of correctness, IBM Laboratory Technical Report TR 25.108, Vienna.