William Worthington Pugh Jr.

1 PERSONAL INFORMATION

EDUCATION

- PhD, Computer Science, Cornell University, 1988 (with a minor in Acting).
- BS, *cum laude*, with honors in Computer Science, Syracuse University, 1980.

EMPLOYMENT

- Professor Emeritus, Department of Computer Science, UMCP.

- Visiting Scientist, Google.
  June 2000 - August 2000, July 2006 - August 2006, July 2007 - August 2007, August 2008 - June 2009, June 2010 - August 2010.January 2011 - *present.*
- (part time) Program Officer, National Science Foundation, January 2011 - August 2011.
- Professor, Department of Computer Science, UMCP.
  Appointed August 2001 — January 2012
- Associate Professor, Department of Computer Science, UMCP.
  Appointed August 1994 — August 2001.
- Associate Professor, Institute for Advanced Computer Studies, UMCP.
  Appointed August 1994 — *present.*
- Assistant Professor, Institute for Advanced Computer Studies, UMCP.
  Appointed August 1989 — August 1994.
- Assistant Professor, Computer Science Department, UMCP.
  Appointed August 1988 — August 1994,
- Software Engineer, Texas Instruments, Dallas, Texas, 1980-1983.

2 RESEARCH, SCHOLARLY and CREATIVE ACTIVITIES

ARTICLES IN REFEREED JOURNALS

1. Skip Lists: A Probabilistic Alternative to Balanced Trees,
   William Pugh,
   Communications. of the ACM, Vol. 33, No. 6, June 1990, pages 668–676

2. Slow Optimally Balanced Search Strategies vs. Cached Fast Uniformly Balanced Search Strategies,
   William Pugh,
   Information Processing Letters, Vol. 34, No. 5, May 1990, pages 251–254

3. The Omega test: a fast and practical integer programming algorithm for dependence analysis,
   William Pugh,
   Communications of the ACM, Vol. 35, No. 8, August 1992, pages 102–114

4. A Partial Evaluator for the Maruti Real-Time System,
   Vivek Nirkhe and William Pugh,
   Real-Time Systems Journal, Vol. 5, No. 1, March 1993, pages 13–30

5. Definitions of Dependence Distance,
   William Pugh,
   Letters on Programming Languages and Systems, Vol. 1, No. 3, September 1993, pages 261–266

6. Parallel Finite Automata for Modeling Concurrent Software Systems
   David Stotts and William Pugh,
   Journal of Systems and Software, (Elsevier Science),
   Vol. 27, No. 1, Oct 1994, pages 27–43.

7. Static Analysis of Upper and Lower Bounds on Dependences and Parallelism,
   William Pugh and Dave Wonnacott,
   ACM Transactions on Programming Languages and Systems,
   Vol. 16, No. 4, July, 1994, pages 1248–1278

8. Determining Schedules based on Performance Estimation,
   Wayne Kelly and William Pugh,
   Parallel Processing Letters, Vol. 4, No. 3, Sept. 1994, pages 205–219.

9. Going beyond Integer Programming with the Omega Test to Eliminate False Data Dependences,
   William Pugh and Dave Wonnacott,
   IEEE Transactions on Parallel and Distributed Systems, Vol. 6, No. 2, Feb 1995, pages 204–211.

10. Parametric Dispatching of Hard Real-Time Tasks,
    Richard Gerber, William Pugh and Manas Saksena,
    IEEE Transactions on Computers, Vol. 44, No. 3, March 1995, pages 471–479.

11. Using Affine Closure to find Legal Reordering Transformations,
    Wayne Kelly and William Pugh,
    International Journal of Parallel Programming, Vol. 23, No. 4, August 1995, pages 303–325.

12. Efficient Distribution Analysis via Graph Contraction,
    Thomas J. Sheffler, Robert Schreiber, William Pugh, John R. Gilbert and Siddhartha Chatterjee,
    International Journal of Parallel Programming, Vol. 24, No. 6, December 1996, pages 599–620

13. Transitive Closure of Infinite Graphs and Its Applications,
Wayne Kelly, William Pugh, Evan Rosser and Tatiana Shpeisman
International J. of Parallel Programming, Vol. 24, No. 6, December 1996, pages 579–598

14. Constraint-Based Array Dependence Analysis
William Pugh and David Wonnacott
ACM Transactions on Programming Languages and Systems, Vol. 20, No. 2, May 1998, pages 635–678

15. Model Checking Concurrent Systems with Unbounded Integer Variables: Symbolic Representations, Approximations and Experimental Results,
Tevfik Bultan, Richard Gerber and William Pugh,
ACM Transactions on Programming Languages and Systems, Vol. 21, No. 4, July 1999, pages 747–789

16. The Java Memory Model is Fatally Flawed,
William Pugh,
Concurrency: Practice and Experience, Vol. 12, No. 6, May 2000, pages 445–455.

REFEREED CONFERENCE PROCEEDINGS

1. ALEX - an Alexical Programming Language,
Dexter Kozen, Tim Teitelbaum, Wilfred Chen, John Field, William Pugh and Brad Vander Zanden,
Proceedings of the 1987 Workshop on Visual Languages,
Linkoping, Sweden, August 19-21, 1987, pages 315-329

2. An Improved Replacement Strategy for Function Caching,
William Pugh,
ACM Conference on Lisp and Functional Languages, July 1988, pages 269-276
(32 papers accepted out of 145 papers submitted)

3. Incremental Computation via Function Caching,
William Pugh and Tim Teitelbaum,
1989 ACM Conference on the Principles of Programming Languages, January 1989, pages 315-328,
(30 papers accepted out of 191 papers submitted)

4. Skip Lists: A Probabilistic Alternative to Balanced Trees,
William Pugh,
Workshop on Algorithms and Data Structures, Ottawa, Canada, August 1989, pages 437-449

5. Two directional record layout for multiple inheritance,
William Pugh and Grant Weddell,
1990 ACM SIGPLAN Conference on Programming Language Design and Implementation, June 1990, pages 85-91
(30 papers accepted out of 173 papers submitted)

6. Probabilistic analysis of set operations using binary hash tries with constant-time set equality testing,
William Pugh,
ICCI-90, May 1990, pages 62-71

7. Application of Partial Evaluation to Hard Real-Time Programming,
Vivek Nirkhe and William Pugh,
Eighth IEEE Workshop on Real-Time Operating Systems and Software and 17th IFAC/IFIP Workshop on Real-Time Programming, May 15-17, 1991, pages 71-75

8. Uniform Techniques for Loop Optimization,
   William Pugh,
   International Conference on Supercomputing, Cologne, Germany, June 17-21, 1991, pages 341-352
   (46 out of 107 papers accepted)

9. The Omega test: a fast and practical integer programming algorithm for dependence analysis,
   William Pugh,
   Supercomputing '91, November, 1991, pages 4-13

10. A Partial Evaluator for the Maruti Hard Real-Time System,
    Vivek Nirkhe and William Pugh,
    12th IEEE Real-Time Systems Symposium,
    December 1991, pages 13-30

11. Partial Evaluation of High-Level Imperative Programming Languages, with Applications in Hard Real-Time Systems,
    Vivek Nirkhe and William Pugh,
    1992 ACM Conference on Principles of Programming Languages,
    January 1992, pages 269-280

12. Eliminating False Data Dependences using the Omega Test,
    William Pugh and David Wonnacott,
    1992 ACM SIGPLAN Conference on Programming Language Design and Implementation,
    June, 1992, pages 140-151

13. An Evaluation of Exact Methods for Analysis of Value-based Array Data Dependences,
    William Pugh and David Wonnacott,
    Sixth Annual Languages and Compilers for Parallelism Workshop,
    August, 1993

14. Counting Solutions to Presburger Formulas: How and Why,
    William Pugh,
    1994 ACM SIGPLAN Conference on Programming Language Design and Implementation,
    June, 1994, pages 121-134.

15. Experiences with Constraint-based Array Dependence Analysis,
    William Pugh and David Wonnacott,
    Principles and Practice of Constraint Programming Workshop,
    May, 1994

16. Simplifying Polynomial Constraints Over Integers to Make Dependence Analysis More Precise,
    Vadim Maslov and William Pugh,
    CONPAR 1994,
    September, 1994

17. Finding Legal Reordering Transformations using Mappings,
    Wayne Kelly and William Pugh,
    Seventh Annual Languages and Compilers for Parallelism Workshop,
    August, 1994

18. Code Generation for Multiple Mappings,
    Wayne Kelly, William Pugh and Evan Rosser,
    5th Symposium on the Frontiers of Massively Parallel Computation,
    February, 1995

19. A Unifying Framework for Iteration Reordering Transformations,
Wayne Kelly and William Pugh, *Proceedings of the IEEE First International Conference on Algorithms And Architectures for Parallel Processing*, April 1995

20. Nonlinear Array Dependence Analysis,
William Pugh and David Wonnacott,
Third Workshop on Languages, Compilers, and Run-Time Systems for Scalable Computers, May 1995

21. Transitive Closure of Infinite Graphs and Its Applications,
Wayne Kelly, William Pugh. Evan Rosser and Tatiana Shpeisman,
Eighth Annual Workshop on Programming Languages and Compilers for Parallel Computing, August 1995

22. Efficient Distribution Analysis via Graph Contraction,
T. J. Sheffler, R. Schreiber, W. Pugh and J. R. Gilbert and S. Chatterjee,
Eighth Annual Workshop on Programming Languages and Compilers for Parallel Computing, August 1995

23. Minimizing Communication while Preserving Parallelism, Wayne Kelly and William Pugh, *International Conference on Supercomputing*, May 1996

24. Exploiting monotone convergence functions in parallel programs,
William Pugh, Evan Rosser, and Tatiana Shpeisman,
Ninth International Workshop on Languages and Compilers for Parallel Computing, San Jose, CA, August 1996.

25. Iteration Space Slicing and Its Application to Communication Optimization,
William Pugh and Evan Rosser,
International Conference on Supercomputing, July 1997, Vienna, Austria.

26. Symbolic Model Checking of Infinite State Systems Using Presburger Arithmetic,
Tevfik Bultan, Richard Gerber and William Pugh,
Proceedings of the 9th International Conference on Computer Aided Verification (CAV '97), Orna Grumberg, ed., LNCS 1254, pp. 400-411, Springer, Haifa, Israel, June 1997.

27. SIPR: A new framework for generating efficient code for sparse matrix computations,
William Pugh and Tatiana Shpeisman,
Eleventh International Workshop on Languages and Compilers for Parallel Computing, Chapel Hill, North Carolina, August, 1998.

28. Compressing Java Class Files,
William Pugh,
ACM SIGPLAN Conference on Programming Language Design and Implementation,
May, 1999

29. Fixing the Java Memory Model,
William Pugh,
ACM Java Grande Conference June, 1999

30. Iteration Space Slicing for Locality,
William Pugh and Evan Rosser,
12th *International Workshop on Languages and Compilers for Parallel Computing* August, 1999

31. More Efficient Network Class Loading through Bundling,
David Hovemeyer and William Pugh,
USENIX Java Virtual Machine Research and Technology Symposium,
May 2001

32. Core Semantics of Multithreaded Java,
Jeremy Manson and William Pugh,
ACM Java Grande Conference June 2001

33. Atomic Instructions in Java,
David Hovemeyer, William Pugh and Jaime Spacco,
European Conference on Object Oriented Programming, June 2002

34. Evaluating the Impact of Programming Language Features on the Performance of Parallel Applications on Cluster Architectures,
Konstantin Berlin, Jun Huan, Mary Jacob, Garima Kochhar, Jan Prins, William Pugh, P. Sadayappan, Jaime Spacco, and Chau-Wen Tseng.
The 16th International Workshop on Languages and Compilers for Parallel Computing, October, 2003.

35. MPJava: High-Performance Message Passing in Java using Java.nio,
William Pugh and Jaime Spacco,
The 16th International Workshop on Languages and Compilers for Parallel Computing, October, 2003.

36. Sequential Consistency is a bad programming model for programmers,
William Pugh,
Dagstuhl seminar on Hardware and Software Consistency Models: Programmability and Performance, October, 2003.

37. The Java Memory Model, Jeremy Manson and William Pugh,
Dagstuhl seminar on Hardware and Software Consistency Models: Programmability and Performance, October, 2003.

38. Finding Concurrency Bugs in Java,
David Hovemeyer and William Pugh,
Proceedings of the PODC Workshop on Concurrency and Synchronization in Java Programs, St. John's, Newfoundland, Canada, July 25-26, 2004.

39. RUBiS Revisited: Why J2EE Benchmarking is Hard,
Jaime Spacco and Bill Pugh,
Component And Middleware Performance Workshop at OOPSLA '04, October 2004

40. An Eclipse-Based Course Project Snapshot and Submission System,
Jaime Spacco, David Hovemeyer and Bill Pugh,
Eclipse Technology eXchange (ETX) Workshop at OOPSLA '04, October 2004

41. Finding Bugs is Easy,
David Hovemeyer and William Pugh. *OOPSLA Onward! track*, October 2004

42. The Java Memory Model,
Jeremy Manson, William Pugh and Sarita Adve,
ACM Conference on the Principles of Programming Languages, 2005

43. Evaluating and Tuning a Static Analysis to Find Null Pointer Bugs,
David Hovemeyer, Jaime Spacco, and William Pugh,
Proceedings of the 2005 ACM SIGPLAN-SIGSOFT Workshop on Program Analysis for Software Tools and Engineering (PASTE 2005), Lisbon, Portugal, September, 2005.

44. Software Repository Mining with Marmoset: An Automated Programming Project Snapshot and Testing System,
Jaime Spacco, Jaymie Strecker, David Hovemeyer, and William Pugh,
Proceedings of the Mining Software Repositories Workshop (MSR 2005), St. Louis, Missouri, USA, May, 2005.

45. What Do High-Level Memory Models Mean for Transactions?,
Dan Grossman, Jeremy Manson and William Pugh,
ACM SIGPLAN Workshop on Memory Systems Performance and Correctness, San Jose, CA, October 2006.

46. Finding More Null Pointer Bugs, But Not Too Many,
David Hovemeyer and William Pugh,
Proceedings of the 2007 ACM SIGPLAN-SIGSOFT Workshop on Program Analysis for Software Tools and Engineering (PASTE 2007), San Diego, CA, June, 2007.

47. Evaluating Static Analysis Defect Warnings On Production Software,
Nathaniel Ayewah, William Pugh, J. David Morgenthaler, John Penix, and YuQian Zhou,
Proceedings of the 2007 ACM SIGPLAN-SIGSOFT Workshop on Program Analysis for Software Tools and Engineering (PASTE 2007), San Diego, CA, June, 2007.

48. Improving software quality with static analysis,
Jeffrey S. Foster, Michael W. Hicks and William Pugh,
PASTE '07: Proceedings of the 7th ACM SIGPLAN-SIGSOFT workshop on Program analysis for software tools and engineering,
San Diego, California, USA, June, 2007.

49. Unit testing concurrent software,
William Pugh and Nathaniel Ayewah,
ASE '07: Proceedings of the twenty-second IEEE/ACM international conference on Automated software engineering,
Atlanta, Georgia, 2007

50. Nathaniel Ayewah, William Pugh, J. David Morgenthaler, John Penix and YuQian Zhou,
Using FindBugs on production software,
OOPSLA '07: Companion to the 22nd ACM SIGPLAN conference on Object oriented programming systems and applications companion,
Montreal, 2007

51. Nathaniel Ayewah and William Pugh,
A report on a survey and study of static analysis users,
DEFECTS '08: Proceedings of the 2008 workshop on Defects in large software systems, July 2008

52. Nathaniel Ayewah and William Pugh,
Learning from defect removals,
MSR '09: Proceedings of the 2009 6th IEEE International Working Conference on Mining Software Repositories, May 2009

53. Nathaniel Ayewah and William Pugh,
Using checklists to review static analysis warnings,
DEFECTS '09: Proceedings of the 2nd International Workshop on Defects in Large Software Systems: Held in conjunction with the ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA 2009), June 2009

54. Nathaniel Ayewah and William Pugh,
Null Dereference Analysis in Practice,
9th ACM SIGPLAN-SIGSOFT Workshop on Program Analysis for Software Tools and Engineering (PASTE), June 2010

55. Nathaniel Ayewah and William Pugh,
The Google FindBugs Fixit,
ISSTA 2010, International Symposium on Software Testing and Analysis July 2010

TALKS

Invited Talks

- A Unified Framework for Reordering Transformations
  Wayne Kelly and William Pugh
  Schloss Dagstuhl Workshop on Parallelization Techniques for Uniform Algorithms
  June 21-25, 1993

- The Omega Library
  William Pugh
  Schloss Dagstuhl Workshop on Loop Parallelization
  April 15-19th, 1996

- Is Code Optimization Relevant?
  William Pugh
  Schloss Dagstuhl Workshop on Code Optimisation: Trends, Challenges and Perspectives
  September 17th-22nd, 2000

- Memory Models for Programming Languages
  William Pugh
  Workshop on Formal Specification and Verification Methods for Shared Memory Systems
  October 31st, 2000

- Is Code Optimization Relevant?
  William Pugh,
  IBM Programming Language Research Day,
  April 26nd, 2001

- Finding Bugs is Easy
  William Pugh,
  Mid-Atlantic Student Workshop on Programming Languages and Systems,
  Saturday, April 26th, 2003

- Judging Static Analysis,
  William Pugh,
  Static Analysis Summit II, November, 2007

- Academic researchers need to listen to their customers, William Pugh, 1st International Workshop on Learning From eXperience (Co-located with PLDI), June, 2010

Developers Presentations

- Using Static Analysis to Find Bugs in Java Software,
  William Pugh,
  The ServerSide 2007 developer conference

- Joshua Bloch, William Pugh,
  Java Puzzlers Episode VI: The Phantom-Reference Menace/Attack of the Clone/Revenge of the Shift
  JavaOne 2007 Developers conference,
  received Java Rockstar award

- Testing Concurrent Software,
  Cliff Click, Brian Goetz, William Pugh,
  JavaOne 2007 Developers conference,
- Improving Software Quality with Static Analysis
  William Pugh,
  JavaOne 2007 Developers conference,
  received Java Rockstar award
- Defective Java Code: Turning WTF Code Into a Learning Experience,
  William Pugh,
  JavaOne 2008 Developers conference
- Using FindBugs in Anger,
  William Pugh,
  JavaOne 2008 Developers conference

Tutorials

- Using FindBugs for Research,
  David Hovemeyer, William Pugh,
  ACM SIGPLAN conference on Programming Language Design and Implementation, 2007
- Using FindBugs for Research,
  David Hovemeyer, William Pugh,
  2007

BOF Presentations

- *The Java Memory Model*, OOPSLA, 1999
- *Revising the Java Thread Specification*, JavaOne, 2000

Research Seminars

- Princeton, *The Omega project*, July 1st, 1997
- Microsoft Research, *The Omega project*, Oct 23rd, 1998
- IBM Watson, *The Java Memory Model*, Feb 24th, 1999
- Sun East, *The Java Memory Model*, July 7th, 1999
- UCSD, *The Omega Project and Iteration Space Slicing*, August 2nd, 1999
- Sun West, *The Java Memory Model*, August 3rd, 1999
- Dec SRC, *The Java Memory Model*, August 3rd, 1999
- MIT, *The Java Memory Model*, Sept 10th, 1999
- Deleware, *The Java Memory Model*, Oct 6th, 1999
- Rutgers, *The Java Memory Model*, Oct 28th, 1999

- UC Berkeley, *The Java Memory Model*, Jan 10th, 2000
- MIT, *The Java Memory Model*, March 27th, 2000

PROFESSIONAL OUTREACH

- Presentation on *Correct and Efficient Synchronization of Java Threads*
  Doug Lea and William Pugh
  JavaOne Developer Conference (25,000 attendees),
  talk attendence more than 3,000, June 6-9, 2000
- Presentation on *Threads, Concurrency and Synchronization*
  David Holmes, Doug Lea and William Pugh
  JavaOne Developer Conference,
  talk attendence more than 1,500, March 28, 2002

RESEARCH SOFTWARE

- *Skip Lists*, a probabilistically balanced data structure. Implementations of skip lists in both `C` and `Pascal` are available via anonymous ftp from `ftp.cs.umd.edu:pub/skipLists`.
- *The Omega test*, algorithms and software for advanced data dependence tests and verification and simplification of Presburger formulas.
- *Extended* `tiny`, an extended version of Michael Wolfe's `tiny` tool.

  Both the Omega test and extended `tiny` are available via anonymous ftp from `ftp.cs.umd.edu:pub/omega`. Our implementations consist of 25,000 lines of code on top of 20,000 lines of code written by Michael Wolfe.
- *FindBugs*, an open source tool for finding defects in Java programs. Downloaded more than 1,300,000 times as of Oct, 2011.
- *MultithreadedTestcase*, an open source tool for testing concurrent abstracts written in Java.
- *Marmoset*, an open source tool for managing the submission, testing and review of student programming projects.

RESEARCH EXHIBITS

- Advanced Program Analysis and Transformations using the Omega test, *Supercomputing*, November 1992, William Pugh, Udayan Borkar, Wayne Kelly, Vadim Maslov, Jerry Sobieski, David Wonnacott

  I prepared the exhibit and demonstration, helped organized the logistics, and, along with other members of my research group, gave demo's. *Supercomputing* is a annual academic and trade conference with 4000+ attendees.
- Implementing OO Languages under a Weak Memory Order, *OOPSLA* 1999 poster session

THEATRICAL PERFORMANCES

- *Biedermann and the Firebugs: A learning-play without a lesson*,
  by Max Frisch,
  adapted by Mordecai Gorelik,
  produced by Cornell University Theater Department,
  performed role of THE LEADER OF THE CHORUS OF FIREMEN,
  Fall 1985

- *Friends*,
  by Kobe Abe,
  produced by Cornell University Theater Department,
  performed role of MAN,
  Spring 1986

- *The Stage You See Before You Is Her Head: Voices of Polish Resistance Theater in 33 Extravagant Scenes, with Clowns, Cabaret, Puppets, Old Time Radio Drama and Lowdown Jazz*,
  adapted and directed by Kim Langford,
  produced by the First Street Playhouse, Ithaca, New York,
  performed roles of: LEADER, JEAN ODEROUS, LORD HAMILTON, ALOYSIUS PTARMIGAN, MALE MANNEQUIN 1, NARRATOR, ARNAUD, PRINCIPLE, ORPHEUS, SEVEN BROTHERS, FUNERAL DIRECTOR, GRAVEDIGGER, THE PUBLIC, LEVOISIN AND GENTLEMAN 3,
  Fall 1986.

CONTRACTS AND GRANTS

- Incremental Computation and Stable Decompositions, *General Research Board of Graduate Studies and Research, University of Maryland*, William Pugh, June-Sept. 1989, $4,750.

- Incremental Evaluation Techniques for Advanced Programming Environments, *National Science Foundation*, William Pugh, Sept 1989 - May 1991, $65,730.

- NSF Presidential Young Investigator Award, *National Science Foundation*, William Pugh, 1991-1996, $312,500.

- Packard Fellowship, *Packard Foundation*, William Pugh, 1991-1996, $500,000.

- NSF Software Capitalization Grant for Software for Array Data Dependence Analysis and Manipulating Sets of Linear Constraints over Integer Variables, *National Science Foundation*, William Pugh, $30,421, 1994-1995.

- Constraint-Based Analysis of Infinite-State Programs, *National Science Foundation*, Rich Gerber and William Pugh, 1997-1999, $ 135,000.

- Advanced Algorithms for the Compilation of Scientific Programs, *National Science Foundation*, William Pugh, 1997-2001, $ 254,009.

- Java Performance and Semantics, Collaborative Research Agreement with *Sun Microsystems*, William Pugh, 1999-2000, $ 49,280.

- System Support for Enterprise Application Servers, *National Science Foundation*, Pete Keleher (PI), Jeff Hollingsworth, William Pugh, Nick Rousopoulos (Co-PI's), September 2000 - September 2003, $861,244.

- Parallel programming paradigms for distributed memory and distributed shared memory high performance computers, *National Security Agency*, William Pugh (PI), Chau-Wen Tseng (PI), Siddhartha Chatterjee (co-PI), and Ponnuswamy Sadayappan (co-PI), November 2000 - October 2001, $320,000.

- Parallel programming paradigms for distributed memory and distributed shared memory high performance computers, *National Security Agency*, William Pugh (PI), Chau-Wen Tseng (PI), Jan Prins (co-PI), and Ponnuswamy Sadayappan (co-PI), April 2002 - April 2003, $345,000.

- Java Performance and Semantics, Collaborative Research Agreement with *Sun Microsystems*, William Pugh, 2000-2001, $ 50,000.

- Java Performance and Semantics, Collaborative Research Agreement with *Sun Microsystems*, William Pugh, 2001-2002, $ 30,000.

- Reliable Multithreaded Software, *National Science Foundation*, William Pugh, 2001-2004, $ 200,000.

- Java Performance and Semantics, Collaborative Research Agreement with *Sun Microsystems*, William Pugh, 2004-2005, $ 30,000.

- Dynamic Compositional Middleware Frameworks for Networked Embedded Systems, *Darpa*, Jan Vitek, Jens Palsberg, Tony Hosking, Doug Lea, William Pugh, 2001 - 2004, $3,274,680 ($358,123 to Pugh).

- FindBugs enhancement and maintaince, Fortify Software, Dec 2005 - Dec 2006, $135,000.

- Understanding Software Defects and Defect Detection Tools, Google, April 2006, $60,000.

- Defect detection tools, Collaborative Research Agreement with *Sun Microsystems*, William Pugh, May 2006, $ 16,000.

AWARDS

- General Electric Fellowship, 1983 - 1984

- AT&T Bell Laboratories Scholar, 1984 - 1988

- NSF Presidential Young Investigator, 1991-1996

- Packard Fellow, 1991-1996

REVIEWING ACTIVITIES

Editorial Boards

- Program Committee, *1991 ACM SIGPLAN Conference on Programming Language Design and Implementation* , June 1991.

- Tutorial Chair, *1992 ACM SIGPLAN Conference on Design and Implementation of Programming Languages* June 1992.

- Editorial Board, ACM *Letters on Programming Languages and Systems (LOPLAS)*, April 1993 →

- Program Committee, *1994 ACM Conference on Principles of Programming Languages* , January 1994.

- Program Committee Co-Chair, *ACM SIGPLAN Workshop on Language, Compiler, and Tool Support for Real-Time Systems*, June 1994.

- Program Committee, *Frontiers of Massively Parallel Computing*, 1995
- Program Committee, *Principles of Programming Languages*, 1996
- Program Committee, *International Conference on Parallel Processing*, 1996
- Program Committee, *Principles of Programming Languages*, 1997
- Program Committee, *EUROPAR* 1997 workshop on automatic parallelization and high performance compilers
- Program Committee, *Intercontinental Workshop on Aliasing in Object Oriented Systems*, 1999
- Program Committee, *Object Oriented Programming Languages, Systems and Applications* 2000, 2003
- Program Committee, *Languages and Compilers for Parallel Computing*, 2000-2003
- Program Committee, *SIGPLAN Conference on Programming Language Design and Implementation*, 2001

Reviewing Activities

- *ACM SIGPLAN Conference on Programming Language Design and Implementation*
- *International Conference on Supercomputing*
- *ACM Conference on Principles of Programming Languages*
- *ACM Letters on Programming Languages and Systems (LOPLAS)*
- *International Conference on Supercomputing*
- *Supercomputing* Conference
- *International Parallel Processing Symposium*
- *Journal of Parallel and Distributed Computing*
- *Parallel Processing Letters*
- *IEEE Transactions on Software Engineering*
- *IEEE Transactions on Parallel and Distributed Systems*
- *Real-Time Systems Journal*
- *ACM Doctoral Dissertation Award*, 1993

3 TEACHING and ADVISING

COURSES

i General Courses

Semester	Course	# students	Description
Fall 1991	CMSC 280	38	Overview of Computer Science
Spring 1992	CMSC 330	44	Organization of Programming Languages
Spring 1998	CMSC 299	40	Freshmen/Sophomore Honors Seminar in Computer Science (with Bill Gasarch)
Fall 2004	CMSC 132	29	Object Oriented Programming II
Spring 2006	CMSC 132	29	Object Oriented Programming II

ii **Specialized Courses**

Semester	Course	students	Description
Fall, 1988	CMSC 430	27	Theory of Language Translation
Spring, 1989	CMSC 838P	18	Implementation of Programming Languages and Environments
Fall 1989	CMSC 451	37	Design of Algorithms
Spring 1990	CMSC 451	34	Design of Algorithms
Fall 1990	CMSC 630	14	Theory of Software
Spring 1991	CMSC 651	12	Advanced Algorithms
Fall 1992	CMSC 430	25	Theory of Language Translation
Fall 1992	CMSC 498	4	Algorithms and Puzzle Solving
Spring 1993	CMSC 838P	20	Program Analysis, Understanding and Optimization
Fall 1993	CMSC 818X	25	Departmental Systems, Programming Languages and Databases Seminar
Fall 1993	CMSC 430	25	Theory of Language Translation
Spring 1994	CMSC 430	32	Theory of Language Translation
Fall 1995	CMSC 838P	25	Program Analysis and Understanding
Spring 1996	CMSC 430	35	Theory of Language Translation
Spring 1996	CMSC 498P	25	Advanced authoring for the WWW
Fall 1996	CMSC 430	55	Theory of Language Translation
Spring 1997	CMSC 498P	40	Advanced Object Oriented Programming
Fall 1997	CMSC 430	50	Theory of Language Translation
Spring 1998	CMSC 430	50	Theory of Language Translation
Fall 1998	CMSC 731	20	Implementing Java
Spring 1999	CMSC 433	45	Programming Language Technologies and Paradigms
Fall 1999	CMSC 433	45	Programming Language Technologies and Paradigms
Spring 2000	CMSC 433	45	Programming Language Technologies and Paradigms
Fall 2000	CMSC 731	30	Programming Language Implementation
Spring 2001	CMSC 433	45	Programming Language Technologies and Paradigms
Fall 2002	CMSC 631	30	Program Analysis and Understanding
Spring 2003	CMSC 433	45	Programming Language Technologies and Paradigms
Fall 2003	CMSC 838p	20	Enterprise Applications
Fall 2004	CMSC 132	34	Object Oriented Programming II
Fall 2005	CMSC 433	49	Programming Language Technologies and Paradigms
Fall 2005	CMSC 433	49	Programming Language Technologies and Paradigms
Fall 2006	CMSC 433	29	Programming Language Technologies and Paradigms (with Vibha Sazawal)
Spring 2007	CMSC 298B	5	FindBugs internship
Fall 2007	CMSC 433	31	Programming Language Technologies and Paradigms
Spring 2008	CMSC 132	15	Object Oriented Programming II
Fall 2009	CMSC 132H	30	Object Oriented Programming II
Spring 2010	CMSC 433	40	Programming Language Technologies and Paradigms
Spring 2011	CMSC 132H	21	Honors Object Oriented Programming II
Fall 2011	CMSC 433	44	Programming Language Technologies and Paradigms

iv **Independent Students**

Semester	Course	# students	Description
Spring 1993	CMSC 386	1	Exper. Learning
Fall 1999	CMSC 386	1	Exper. Learning

DEVELOPMENT

CMSC430

- Developed project for converting regular expressions into minimized DFA's. Wrote 700+ lines of flex, bison and C++ code that was provided to students as a base for the project. Project was used by others teaching 430.
- Developed project for reading a grammar, computing the LL(1) lookahead sets and the LALR(1) goto graph. Wrote 950+ lines of flex, bison and C++ code that was provided to students as a base for the project. Project was used by others teaching 430.
- Developed compiler project infrastructure, including symbol table and intermediate format support infrastructure. Wrote 4500+ lines of flex, bison and C++ code, most of which was provided to students as a base on which to build there projects. Infrastructure was used by others teaching 430.

CMSC433

- Developed (and am continuing to develop and evolve) entire course and projects. No appropriate text book exists for the course, so the course material is put together from many different sources. For each semester, the course projects involve a total of about 3000+ lines of C++ and Java that I write, more than half of which is new each semester.

CMSC731

- Led students to develop infrastructure for a Java bytecode optimizer and analyzer.

132

- Designed and taught new intro course as part of a redesign of the introductory computer science course sequence.

Marmoset

- Designed and implemented Marmoset, the project submission and testing framework now used in many CS courses.

AWARDS

- Recognized for mentoring undergraduates by the Dean of Undergraduate Studies (nominated by David Bagett, who entered the PhD program at MIT on an NSF Fellowship) and by the Dean of CMPS (nominated by Winston Cockfield).
- Department of Computer Science Teaching Excellence award for Faculty, May 1997
- Univ. of Maryland Innovation in Teaching with Technology, 2006 (with Jaime Spacco)

RESEARCH DIRECTION

ii **Masters**

- Udayan Borkar (1991 - 1992),

iii **Doctoral**

- Dominic Duggan (1989-1992), now Associate Professor at Stevens Institute of Technology
- Vivek Nirkhe (*with Satish Tripathi*, 1991-1992), now program manager at Microsoft working on Content Manager Server
- Dave Wonnacott (1991 - 1995), Now Associate professor at Haverford
- Wayne Kelly (1992 - 1996), now a senior lecturer at Queensland University of Technology, Brisbane, Australia
- Evan Rosser (1993 - 1998), now working at Synopsys
- Tatiana Shpeisman (1993 - 1999), now working at Intel
- Jeremy Manson (1999 - 2004), now at Google
- David Hovemeyer (1999 - 2005), now an assistant professor at York College
- Jaime Spacco (2001 - 2006), now a visiting faculty member at Colgate College
- Nat Ayewah (2006-2010)

4 SERVICE

PROFESSIONAL

i **Editorial activities**

- Editor-in-Chief of *ACM Transactions on Programming Languages and Systems (TOPLAS)*, Feb 1997 – Jan 2000

ii **Unpaid reviewing activities for agencies**

- NSF Instrumentation award panel, November 1990
- NSF RIA award panel, April 1991
- NSF NYI award panel, April 1993
- NSF award panel, June 1999

iii **Other non-University committees**

- Tutorial Chair, *ACM SIGPLAN'92 Conference on Programming Language Design and Implementation (PLDI)*, San Francisco, California, June 17-19, 1992.
- Conference Co-Chair, *ACM SIGPLAN Workshop on Language, Compiler, and Tool Support for Real-Time Systems*, June 1994.
- Conference Co-Chair, *Workshop on Languages and Compilers for Parallel Computers*, August 2002.
- Conference Chair, *ACM SIGPLAN Conference on Programming Language Design and Implementation*, June 2004
- Conference co-Chair, *ACM SIGPLAN Workshop on the Evaluation of Software Defect Detection Tools*, June 2005

UNIVERSITY

i **Department**

- Software Engineering/Programming Languages Field Committee Chair, Fall 1989 → Spring 1994
- General responsibility for SE/PL comp (generally writing and grading 1/3 – 1/2 of all questions), Fall 1991 → Spring 1994
- Graduate admissions, US students, 1990, 1991, and 1992.
- Advisor to Student ACM group Fall 1990 – Spring 1992.
- Enforced University Honor code, requiring more than 40 hours of time, Fall 1991 – Spring 1992.
- UMIACS Review Panel, Fall 1991– Spring 1992
- Directory of Departmental Honors Program, Fall 1991 – Spring 1992.
- Judge and Organizer, High School Programming Contest, Spring 1991, 1992, 1993.
- Designed and wrote database system for managing faculty applicants, Spring 1992
- Chair, search panel for UMIACS Computer Engineer, Summer 1993
- In charge of establishing departmental Mosaic/WWW/gopher server and electronic distribution and indexing of departmental technical reports, Fall 1993 – Spring 1994.
- Did 1/2 of Fall 1995 SE/PL comprehensive exam

- Faculty webmaster, Fall 1995–
- Search committee for CS librarian, Spring 1996
- Merit Pay Committee, Spring 1996
- Graduate student committee, 1995–1996
- Tenure review committee for Dr. Richard Gerber, Fall 1995
- Handled 1/3rd of American Grad. applicants, Spring 1996
- Taught HTML course for faculty and staff, Jan 1996
- Friday Faculty Lunch coordinator: 1996 – 2001
- CS Department Council, 1997–2000
- Dept. Webmaster, 1997–1999
- Faculty Recruiting committee, Spring 1998
- SE/PL Field committee chair, Fall 1998 – Spring 1999
- CS Merit pay committee, 2000
- US grad student admissions
- CSIC building committee, 1999–2001
- CMPS PPC Representative: 2000 – 2001
- Undergraduate co-director: Feb 2000 – August 2001
- Undergraduate awards committee: 2000–2001
- PhD academic placement committee, 2002 – 2003
- Construct and maintain system for webcasting colloquium talks CSIC, 2002 -2003
- Construct and maintain software system for plasma displays in CSIC, 2002 -
- Construct and maintain marmoset system for automated programming project grading.
- Chair of APT committee for Jeff Hollinsworth, Fall 2005
- Chair of Introductory Programming Course committee, Fall 2005

ii **College**

- Discussed attending graduate school with current undergraduates, Fall 1990, 1991, 1992, Spring 1993, Fall 1993.
- UMIACS review panel
- Gave HTML seminar for campus, Feb 1996
- Gave Java seminar for campus (twice), Feb 1996

iii **Campus**

- Member, Graduate Council, 1997–1999

COMMUNITY

- Donated bone marrow for 29 year old woman with AML Leukemia, August 14th, 2000 (matched through national bone marrow registry).

I certify that this CV is accurate and complete — Nov 27th, 2012

William Worthington Pugh Jr.