

Owen L. Astrachan

Department of Computer Science
Box 90129
Duke University
Durham, NC 27708-0129
telephone: (919) 660-6522
email: ola@cs.duke.edu

January 8, 2016

I. Education

Ph.D.	Computer Science	Duke University	1992
M.S.	Computer Science	Duke University	1989
M.A.T.	Mathematics	Duke University	1979
A.B.	Mathematics	Dartmouth College	1978

with distinction in Mathematics, Summa Cum Laude, Phi Beta Kappa

II. Professional Appointments

Duke University, Department of Computer Science

Professor of the Practice

July 2000 — .

Associate Professor of the Practice

July 1996 — July 2000.

Director of Undergraduate Studies

September 1993 — .

Assistant Professor of the Practice of Computer Science

1993 — 1996.

Lecturer in Computer Science

1991–1992. Developed and taught the introductory course for non-majors. Served on lab committee determining priorities for physical improvements.

Research Assistant

1991 (June–Aug). Research Assistant at SRI International, AI group, working for Mark Stickel on the design of intelligent and efficient automated reasoning systems.

Research Assistant

1988–1991. Research Assistant for Donald W. Loveland, investigating automated theorem proving. Designed and implemented an OR parallel theorem prover that runs on a BBN Butterfly GP-1000, TC2000 and on a network of Sun workstations.

Senior Graduate Instructor

1986–1988. Solely responsible for developing the curriculum and teaching the first course for majors in the Computer Science Department. Assisted with course in Operating Systems.

Teaching Assistant

1985–1986 Served as Teaching Assistant for the the first two courses for majors in Computer Science. Responsible for designing laboratory exercises and running recitation sections.

University of British Columbia, Computer Science Department

Visiting Scholar and Lecturer

Sept 1998 — June 1999 (on sabbatical from Duke)

Experience in Secondary Education

Math and Computer Science Teacher Durham Academy

1980–1985. Taught Advanced Placement Calculus, Advanced Placement Computer Science, Multivariable Calculus and Linear Algebra, Geometry, Introduction to Computer Programming, Finite Mathematics. Developed curriculum for Finite Math, AP Computer Science, and Multivariable Calculus.

Math Teacher Camp Lejeune HS, Camp Lejeune, NC

1978–1980. Taught honors Trigonometry, honors Geometry, Algebra I, Pre-Algebra.

III. Honors

2007, NSF, CISE Distinguished Education Fellow, *Interdisciplinary Problem- and Case-based Computer Science*, one of two inaugural CDEF awardees (see grants).

2004, IBM Faculty Award, *Issues in Large-scale Software Componentization*

2003, ACM International Collegiate Programming Contest (ICPC) Coaches Award.

2002, Richard K. Lublin Award for Distinguished and Motivating Teaching

2002, Nominated for Alumni Distinguished Teaching Award

2001, Nominated for Alumni Distinguished Teaching Award

1998, Outstanding instructor of Computer Science, University of British Columbia (teaching on sabbatical)

1997, NSF Career Award

1996, Nominated for Alumni Distinguished Teaching Award

1995, Duke University, Trinity College of Arts and Science: Robert B. Cox Distinguished Teaching in Science Award

1995, Sigma Xi

1994, Nominated for Alumni Distinguished Teaching Award

1978, A.B. degree awarded with distinction, summa cum laude, Phi Beta Kappa

IV. Publications

Journals:

Steve Wolfman, Owen Astrachan, Mike Clancy, Kurt Eiselt, Jeffrey Forbes, Diana Franklin, David Kay, Mike Scott, and Kevin Wayne. Teaching-Oriented Faculty at Research Universities. *Communications of the ACM*. November 2011, v. 54. n 11. pp. 35-37.

Owen Astrachan and Robert Dewar. CS Education in the U.S.: Heading in the Wrong Direction. *Communications of the ACM*. July 2009, v. 52, n. 7, pp. 41-45.

O.L. Astrachan and D.W. Loveland. The Use of Lemmas in the Model Elimination Procedure. *Journal of Automated Reasoning*, v. 19 n.1, August, 1997, pp. 117-141.

Owen Astrachan, Kim Bruce, Robert Cupper, Peter Denning, Scot Drysdale, Tom Horton, Charles Kelemen, Cathy McGeoch, Yale Patt, Viera Proulx, Roy Rada, Richard Rasala, Eric Roberts, Steven Rudich, Lynn Stein, Allen Tucker, Charles van Loan. Strategic Directions in Computer Science Education. *ACM Computing Surveys*. v 28, n 4, December 1996.

O.L. Astrachan. METEOR: Exploring Model Elimination Theorem Proving. *Journal of Automated Reasoning*. v.13 n.2, 1994, pp. 283-296.

Books:

Owen Astrachan. *A Computer Science Tapestry: Exploring Programming and Computer Science with C++*, Second edition. McGraw-Hill, 2000.

Owen Astrachan. *A Computer Science Tapestry: Exploring Programming and Computer Science with C++*. McGraw-Hill, 1997.

Owen Astrachan. *The Large Integer Case Study in C++*. The College Board, Advanced Placement Program, 1997.

Book Chapters:

Owen Astrachan and Robert Duvall and Eugene Wallingford. Bringing Extreme Programming to the Classroom. in *Extreme Programming Perspectives*, Giancarlo Succi (ed.), Addison Wesley, 2002.

O.L. Astrachan and D.W. Loveland. METEORs: High Performance Theorem Provers Using Model Elimination. in *Automated Reasoning: Essays in Honor of Woody Bledsoe* ed. R.S. Boyer, Kluwer Academic Press 1991.

Miscellaneous:

O.L. Astrachan and Susan Horwitz and the Advanced Placement Computer Science Development Committee. *Communications of the ACM*. Technical Opinion: The First Course Conundrum. June, 1995. Pages 117-118.

Refereed Conferences:

Owen Astrachan, R. Brook Osborne *et al.* Computer Science Principles: Analysis of a Proposed Advanced Placement Course. *SIGCSE Technical Symposium on Computer Science Education*, Denver, CO, 2013

Owen Astrachan. Pander to Ponder, *SIGCSE Technical Symposium on Computer Science Education*, Chattanooga, TN, 2009

Casey Alt, Owen Astrachan, Jeffrey Forbes, Richard Lucic, and Susan Rodger. Social Networks Generate Interest in Computer Science. *SIGCSE Technical Symposium on Computer Science Education*, Houston, TX, 2006.

Owen Astrachan. Non-Competitive Programming Contest Problems as the Basis for Just-in-time Teaching. *Proceedings of Frontiers in Education*, October 2004.

Owen Astrachan. Bubble Sort: An Archaeological Algorithmic Analysis. *SIGCSE Technical Symposium on Computer Science Education*, Reno, NV, 2003.

Owen Astrachan and David Bernstein and Andrew English and Benjamin Koh. Development Issues for a Networked, Object Oriented Gaming Architecture (NOOGA) Teaching Tool. *Proceedings of Frontiers in Education*, November 2002.

Owen Astrachan and Robert Duvall and Jeffrey Forbes and Susan Rodger. Active Learning in Small to Large Courses *Proceedings of Frontiers in Education*, November 2002.

Owen Astrachan and Robert Duvall and Eugene Wallingford. Bringing Extreme Programming to the Classroom, *Proceedings of XPUniverse*, Raleigh, NC, July, 2001.

Owen Astrachan. OO Overkill: When Simple is Better than Not, *SIGCSE Technical Symposium on Computer Science Education*. Charlotte, NC, February 2001.

Charles Keleman, Allen Tucker, Peter Henderson, Kim Bruce, Owen Astrachan. Has Our Curriculum Become Math-Phobic?, *SIGCSE Conference on Integrating Technology into Computer Science Education (ITiCSE)*, June 2000.

Owen Astrachan and Eugene Wallingford. Loop Patterns. *Pattern Languages of Programming (PLoP)*, Allerton Park, IL, August, 1998.

Owen Astrachan. Hooks and Props as Instructional Technology. *SIGCSE Conference on Integrating Technology into Computer Science Education (ITiCSE)*, August 1998.

- Owen Astrachan, Geoffrey Berry, Landon Cox and Garrett Mitchener. Design Patterns: An Essential Component of CS Curricula. *SIGCSE Technical Symposium on Computer Science Education*. Atlanta, GA, February 1998.
- Owen Astrachan and Susan Rodger. Animation, Visualization, and Interaction in CS 1 Assignments, *SIGCSE Technical Symposium on Computer Science Education*. Atlanta, GA, February 1998.
- Owen Astrachan and Robert Smith and James Wilkes. Application-based Modules using Apprentice Learning for CS 2. *SIGCSE Technical Symposium on Computer Science Education*. San Jose, CA, February 1997, pp 233–237.
- Owen Astrachan and Trevor Selby and Joshua Unger. An Object-Oriented, Apprenticeship Approach to Data Structures using Simulation. *Frontiers in Education*, Salt Lake City, Utah, 1996, pp 130–134.
- Owen Astrachan and David Reed. AAA and CS1 : The Applied Apprenticeship Approach to CS 1. *SIGCSE Technical Symposium on Computer Science Education*. Nashville, TN, March 1995.
- Owen Astrachan and Claire Bono. Using simulation in an objects-early approach to CS1 and CS2. *OOPSLA Conference Proceedings, Educator's Forum*. Portland, Oregon, October 1994.
- O. L. Astrachan and D.W. Loveland. METEOR: Model Elimination Theorem Proving with Lemmas (system abstract). *Twelfth Conference on Automated Deduction (CADE-12)*. Nancy, France, 1994.
- Owen Astrachan. Self reference is a Thematic Essential. *SIGCSE Technical Symposium on Computer Science Education*. Phoenix, Arizona, March 1994.
- Owen Astrachan. METEOR: Exploring Model Elimination Theorem Proving. *Workshop on Theorem Proving with Analytic Tableaux and Related Methods*. Marseilles, France, April 1993.
- Owen L. Astrachan, Vivek Khera, and David Kotz. The Duke Internet Programming Contest: A Report and Philosophy. *SIGCSE Technical Symposium on Computer Science Education*. Indianapolis, IN, February 1993.
- Owen L. Astrachan and Mark E. Stickel. Caching and Lemmaizing in Model Elimination Theorem Provers. *Eleventh Conference on Automated Deduction (CADE-11)*. Saratoga Springs, NY, June 1992.
- Owen Astrachan. Finding a Stable roommate, job or spouse: a case study crossing the boundaries of Computer Science Courses. *SIGCSE Technical Symposium on Computer Science Education*. Kansas City, MO, March 1992.
- Owen Astrachan. Pictures as Invariants. *SIGCSE Technical Symposium on Computer Science Education*. San Antonio, TX, March 1991.
- Owen Astrachan. METEOR: Model Elimination Theorem Prover for Efficient OR-Parallelism. *AAAI Spring Symposium on Representation and Compilation in High Performance Theorem Proving: Titles and Abstracts*, ed. W.W. Bledsoe, M. Stickel, P. Lincoln, R. Overbeek, and D. Plaisted, Stanford, CA, March 1989.

Unrefereed Reports:

- Owen L. Astrachan and Donald W. Loveland *The Use of Lemmas in the Model Elimination Procedure*. Duke University Technical Report CS-1993-25.
- Owen L. Astrachan. *METEOR: Exploring Model Elimination Theorem Proving*. Duke University Technical Report CS-1992-22.
- Owen L. Astrachan. *Investigations in Model Elimination Based Theorem Proving*. Ph.D. Thesis. Also Duke University Technical Report CS-1992-21.
- Owen L. Astrachan and Mark E. Stickel. *Caching and Lemmaizing in Model Elimination Theorem Provers*. SRI International Technical Note 513, December 1991.

V. Service

Professional Service

- 2015 Program Committee, Research in Equity and Sustained Participation in Engineering, Computing, and Technology (RESPECT), IEEE, Charlotte, NC.
- 2015 Program Committee, ICER: International Computer Science Education Conference.
- 2014– K-5 Program Affiliate for Code.org Professional Development.
- 2014 Program Committee, ICER: International Computer Science Education Conference.
- 2013 Program Committee, ICER: International Computer Science Education Conference.
- 2013– Code.org Educational Advisory Council.
- 2009 – 2012 *College Board* HEAC: Higher Education Advisory Committee for Advanced Placement. Provide oversight and advice regarding the program.
- 2009, July, NSF Review Panel: Broadening Participation in Computing (BPC)
- 2009 *NITRD: Networking and Information Technology Research and Development Program*, panelist at public forum for discussion of the 2009 Federal Strategic Plan
- 2008 – *NSF/College Board* joint group on the First Year in Computer Science (chair).
- 2008, May, NSF Review Panel *CPATH*
- 2007 – *College Board AP Computer Science Redesign Commission* Committee charged with examining and redefining the Advanced Placement Computer Science Program.
- 2006– ACM Ed-Council. One of 25 members providing leadership and governance to the ACM about educational activities and outreach.
- 2005, July, NSF Review Panel *Advanced Learning Technology*
- Program Committee OOPSLA 2005, Educator’s Symposium*
- Program Committee OOPSLA 2004, Educator’s Symposium*
- Internet & Society Idea Exchange* Faculty Steering Committee for courses related to Internet and Society (oversight from Harvard and MIT, including 55 faculty from around the world).
- NSF Review Panel CRCD Program*
2004, reviewed proposals for the CRCD program in Computer Science at NSF.
- ACM/College Board Digital Library Project* 2004 – Advisory Committee to develop a project supporting Compute Science in high schools as part of the national NSF-sponsored digital library program.
- ACM/College Board JETT Steering Committee* 2002 – , Member of four-person steering committee providing oversight for joint ACM/College Board committee reviewing and approving national sites to host high school outreach programs for computer science.
- Illinois Math and Science Academy*
2002 – 2003, Member of three-person external review board for Mathematics/Computer Science at IMSA.
- College Board/High School Computer Science AP Computer Science*
2001 – 2002, Member of College Board *ad hoc* professional development committee to develop standards for training/educating high school teachers and workshop consultants in computer science.
- NSF Review Panel CISE Program*
2000, reviewed proposals for the CRCD program in Computer Science at NSF.
- Math and Computer Science Mathematical Association of America*
1999. Committee made recommendations to the MAA on the role of mathematics in computer science. Committee consisted of Alan Tucker, Charles Keleman, Dale Skrien, Charles van Loan, Peter Henderson, Kim Bruce, Owen Astrachan.

Chair, Advisory Committee for AP Computer Science College Board

1999–2000. Committee making recommendations on the use of new languages and curricula in Advanced Placement Computer Science. Committee consists of David Gries, Robert (Corky) Cartwright, Henry Walker, Ursula Wolz, Cay Horstmann, Fran Trees, Rich Kick.

External Oversight Board North Carolina Central University

1997 – 1998, oversee the growth and accreditation of the Computer Science Department.

NSF Review Panel CISE Program

1997, reviewed proposals to the Education Innovation program in the CISE directorate of NSF.

External Review Committee Oberlin College

1996, Member of external review committee to evaluate computer science department at Oberlin.

Program Committee CADE-13

1995/6, Member of program committee for CADE-13, Conference on Automated Deduction.

Chair, Advisory Committee for AP Computer Science College Board

1995–1996. Committee makes recommendations on the best use of C++ on the Advanced Placement Exam. The Committee is convened by the College Board, with representation from SIGCSE, the Computer Science Education special interest group of the ACM.

Judge, ACM International Programming Contest Association for Computing Machinery

1994–1997. One of six people responsible for developing problems and judging solutions for the ACM Programming Contest finals.

Chief Reader, Advanced Placement Computer Science Educational Testing Service

1989–1994. Responsible for developing grading standards and assigning scores for the AP exam in Computer Science. Assist with the development of the exam. Oversee, hire, and manage 70 University faculty consultants and High School educators in the grading of 10,000 AP exams taken by secondary students throughout the world.

Member, AP Computer Science Development Committee The College Board

1985–1989. Responsible for developing curriculum and devising tests for the AP exam in Computer Science.

Director, Duke Internet Programming Contest

1990 — 1994 . Co-founded a computer programming contest held in real-time over the Internet, involving 60 teams from 37 institutions in 5 countries (1990); 240 teams from 100 institutions in 9 countries (1991); 290 teams from 140 institutions in 14 countries (1992), 495 teams from 200 institutions in 20 countries (1993). Developed the problems used in the contest, designed solutions for the problems, and co-directed the administration of the contest.

IEEE Programming Contest

1994–1995. Coach of the Duke undergraduate IEEE Programming team. This competition is by invitation only to sixteen teams throughout the world. In 1994 Duke participated for the first time. In 1995 Duke won the contest.

ACM Programming Contest

1993–. Coach of Duke undergraduate ACM Programming Team. In 1994 the team won the mid-Atlantic regional contest and placed third in the world (first U.S. team) in the world finals. In 1995 the team won the mid-Atlantic regional contest and advanced to the world finals, finishing 22nd. In 1997 the team advanced to the world finals. In 1998 the team advanced to the world finals. (On sabbatical in 1999-2000.) In 2001, the team advanced to the world finals. In 2002 Duke had four teams in the top fifteen, and the top two in the Midatlantic regional contest; the top team advances to the world finals. In 2003 Duke advanced to the world finals. In 2004 Duke won the region (tied for first), advanced to the world finals and had three teams in the top 15 of the region; in the world finals Duke was one of four US teams that placed (above honorable mention) and was tied with Caltech and MIT for second among US teams. In 2005 Duke won the Region and advanced to the world finals receiving an honorable mention. In 2006 a Duke team advanced to the world finals, in 2007 Duke won the region and participated in the world finals, in 2008 Duke received an at-large bid to the world finals (to be held in 2009).

1989–1990. Member of Duke Programming team, 1989 and 1990 ACM International Programming Contests. Finished fourth in 1989 (Louisville, KY) and eighth in 1990 (Washington, DC) contest (world) finals.

Referee Activities

2014	SIGCSE	Technical Symposium on Computer Science Education
2013	SIGCSE	Technical Symposium on Computer Science Education
2012	SIGCSE	Technical Symposium on Computer Science Education
2011	SIGCSE	Technical Symposium on Computer Science Education
2010	SIGCSE	Technical Symposium on Computer Science Education
2009	SIGCSE	Technical Symposium on Computer Science Education
2008	SIGCSE	Technical Symposium on Computer Science Education
2007	SIGCSE	Technical Symposium on Computer Science Education
2006	SIGCSE	Technical Symposium on Computer Science Education
2005	ITiCSE	SIGCSE Conference on Integrating Technology into Computer Science Education (ITiCSE)
2005	SIGCSE	Technical Symposium on Computer Science Education
2004	ITiCSE	SIGCSE Conference on Integrating Technology into Computer Science Education (ITiCSE)
2004	SIGCSE	Technical Symposium on Computer Science Education
2003	SIGCSE	Technical Symposium on Computer Science Education
2001	SIGCSE	Technical Symposium on Computer Science Education
2000	SIGCSE	Technical Symposium on Computer Science Education
1996	SIGCSE	Technical Symposium on Computer Science Education
1995/6	CADE-13	Conference on Automated Deduction
1995	TABLEAUX '95	Workshop on Theorem Proving with Analytic Tableaux and Related Methods
	IJCAI	International Joint Symposium on Artificial Intelligence Automated Reasoning Track
	SIGCSE	Technical Symposium on Computer Science Education
1994	SIGCSE	Technical Symposium on Computer Science Education
1993	ILPS	International Logic Programming Symposium
	IJCAI	Automated Reasoning track
	ISMIS	International Symposium on Methodologies for Intelligent Systems
	SIGCSE	Technical Symposium on Computer Science Education
1992	CADE-11	Conference on Automated Deduction
	SIGCSE	Technical Symposium on Computer Science Education
1991	SIGCSE	Technical Symposium on Computer Science Education
1990	CADE-10	Tenth Conference on Automated Deduction

Duke Service

2012-2013. Committee on Facilities and Environment.

2011-2013. Member of Academic Council.

2009. Chair promotion committee for Jeffrey Forbes.

2008. Member of Office Education Committee (OEC) overseeing appointments of ROTC faculty.

2007-2009. Member of Academic Council.

2007-2010. University Committee on Admissions and Financial Aid (Academic Council).

2007–2009 Member of QEP (quality enhancement plan) University Committee to create a 75-page document outlining Duke’s future as part of our 10-year SACS re-accreditation process.

2008 Search Committee, Dean to replace Robert Thompson, resulted in appointment of Lee Baker.

2007. Chair promotion committee for Susan Rodger.

2007-2009. University Commencement Committee.

2006-2007. Chair *ad hoc* Committee to Distinguish Trinity College Degrees. Will report on the status of the BA and BS degrees at Duke.

2006-. Member of Faculty Research Committee that decides on Trinity College competitive grants and awards to faculty for research and conference activity.

2005. Member of Committee on Departmental Support for Technology as part of the University committees on Strategic Directions.

2005. Member of Arts and Science search committee for the A&SIST Associate Dean.

2004-2005. Chair re-appointment committee for Prof. Richard Lucic

2004-. ISIS (Information Sciences and Information Studies) Faculty Steering Committee.

2004-. One of three faculty overseeing the development of teaching and learning, in conjunction with the CIT, as part of re-allocation of resources regarding the former center for teaching and learning.

2004-2007. Member Executive Committee of the Arts and Science Council (ECASC).

2004-2007. Member of ITAC, Committee on Information Technology at Duke.

Chair re-appointment committee for Prof. Richard Lucic 2002.

Member re-appointment committee for Prof. Jeffrey Forbes 2002.

Member Advisory Board for BlackBoard at Duke, 2002-

CITIE, IT skills committee, 2002.

Member Academic Integrity Council, 2001-

Member Executive Committee of the Arts and Science Council (ECASC) 2000-2003.

Interviewed candidates for A.B. Duke program, 2000, 2001

Member Board of Directors for Center for Instructional Technology (CIT), 1999-.

Arts and Science Council, 1999-2008.

Chair, Arts and Science Committee on Integrated Cluster Classrooms, 1999-2000.

Chair re-appointment/promotion committee for Prof. Robert Duvall, 2000.

Chair, University Planning Group on Instructional Technology, 1999-2000.

Search Committee, head of Career Development Center, 1998.

Member B.N. Duke Scholarship Committee, 1997

Member Core Team on evaluating use of Instructional and Information Technology, 1997.

ITAC Committee on Student Computing, 1996.

Chair search committee for Lecturer position in Computer Science, 1996.

Chair re-appointment/promotion committee for Prof. Susan Rodger, 1996.

Search Committee, Assistant Dean of Student Development, 1996.

Arts and Science Council, 1995-1996.

Member of Management Team (Center for Teaching and Learning) to develop an Exercise in Interactive Theatre for "Developing Teacher Knowledge", 1996.

Director of Undergraduate Studies, 1993-.

Provost's ad-hoc committee for Computer Technology and Education, 1995-1997.

Member of Steering Committee, Schulzbegeer Interactive Learning Laboratory, Teaching and Learning Center, 1994-1996.

Faculty Advisor, DULUG: Duke University Linux User's Group, 1995-.

Departmental major advisor 1992- (supervise 20 first majors, 19 second majors per year.)

Premajor advisor at Duke University, 1986- 1998(ten first year students each year, total of 20 per year)

Chaired search committee for Assistant Professor of-the-practice, 1994.

Committee on the role of teaching for graduate students 1990–1992

VI. Consultancies

Coursera Specialization

2015, created five course MOOC, Java Programming, An Introduction to Software Specialization, <https://www.coursera.org/specializations/java-programming>

Summit6 v. Motorola Mobility

2015 (February-June). Retained by Motorola and Google (Kilpatrick Townsend and Stockton), Northern District of Texas, in a patent infringement and invalidity case relating to mms-messaging and sending media over the Internet, 7:14-cv-00014, 7:14-cv-00160. Prepared both invalidity and non-infringement reports, case settled before deposition. Worked primarily with attorney Shayne O'Reilly, also with Jon Harris, Clay Holloway.

CA v New Relic

2013–2014. Retained by CA, Inc (Bracewell and Giuliani), Eastern District of New York, in a patent infringement case related to application performance monitoring, Case 12-cv-05468-JS-WDW. Prepared expert report, deposed in March 2014. Worked with attorneys Barry Shelton, Calvin Cheng, others.

Expert Witness, Greenberg Traurig

2011 — Retained by Google in “Oracle v Google: Complaint for Patent and Copyright Infringement: Java/Android”, US District Court, Northern District of California. Prepared expert report, deposition in matters related to Java/Android APIs and copyright.

Expert Witness, Alston and Bird

2010 — Retained by Nokia in “Certain Mobile Communications and Computer Devices and Components Thereof,” ITC Inv. No. 337-TA-704 (complaint filed Jan. 15, 2010), before the US International Trade Commission. Worked on Markman report, expert report, deposition and testimony.

Expert Witness, Alston and Bird

2008/2009 — Retained by Plaintiffs in Move, Inc., Nat’l Assoc. of Realtors, and Nat’l Assoc. of Home Builders vs. Real Estate Alliance Ltd et al., 2:07-CV-02185-GHK-AJW (filed Apr. 3, 2007) in the Central District of California. Worked on claim construction and expert report preparation.

Consultant, College Board

2008 – Oversee and help plan colloquium for college faculty (attended by 70 faculty) to understand current and future directions for AP Computer Science.

Google

2006 (six months) – Worked as an external contractor to help develop material used internally at Google for educating Google software engineers about Java and C++ programming.

Expert Witness, Womble, Carlyle, Sandridge and Rice

Software expert regarding work related to a contract dispute.

Consultant, AP Computer Science Educational Testing Service

1994 – Advise development committee on incorporating C++ and Java into the Advanced Placement Program. Wrote Pascal/C++ case studies for use on the exam. Provided workshops for high school and college consultants in making transition to object oriented programming. Critique free response questions that are part of the national exam (2002, 2004).

Case-Study author, AP Computer Science Educational Testing Service

1994 – Write the case study for use in the 1997-2000 AP exams. Write the code for the case study used in the 2001-2004 exams. A case is a “literate program”, a treatise on the design, development, and implementation of a programming solution to a problem.

VII. Panels/Conference Activities

- Advanced Placement Computer Science Principles (APCSP): A report from Teachers* with R. Brook Osborne, SIGCSE Conference Memphis, TN, 2016.
- Advanced Placement Computer Science Principles (APCSP): A report from Teachers* with R. Brook Osborne, CSTA Conference, San Diego, CA, 2016.
- Advanced Placement Computer Science: Two Courses, All Students*, with Lien Diaz, CSTA Conference, San Diego, CA, 2016.
- High School CS Breadth in Depth: CSP x ECS x APCSA*, with Gail Chapman and Don Yanek. CSTA Annual Conference, Dallas, TX, 2015.
- Computer Science Principles Curricula*, with Dan Garcia, Bennett Brown, Jeff Gray, Ralph Morelli, Nigamanth Sridhar, and Baker Franke. CSTA Annual Conference, Dallas, TX, 2015.
- Partners for a Path in K-12 Computer Science*, with Pat Yongpradit, Gail Chapman, Joanna Goode, Emmanuel Schanzer, Kiki Prottzman, and Irene Lee. CSTA Annual Conference, Dallas, TX, 2015.
- Computer Science Principles Curricula: On-the-ground, adoptable, adaptable, approaches to teaching*, with Daniel Garcia, Bennett Brown, Jeff Gray, Ralph Morelli, Marie des Jardins, Calvin Lin, Nigamanth Sridhar, Bradley Beth. SIGCSE, Kansas City, MO, 2015.
- Scaling High School Computer Science: Exploring Computer Science and Computer Science Principles* with Ralph Morelli, Jeff Gray, Gail Chapman. SIGCSE, Kansas City, MO, 2015.
- CSP Pilot Instructors: Update from the Classroom*, with Brook Osborne, Lynn Norris, Crystal Furman, and Seth Pizzo. CSTA Annual Conference, St. Charles, IL, 2014.
- Search: Programming and Computational Thinking* CSTA Annual Conference, St. Charles, IL, 2014.
- AP Computer Science Principles: A New AP Course and Exam*, with Rich Kick, Lien Diaz. National Council of Teachers of Mathematics (NCTM), New Orleans, LA, April 2014.
- Diverse Learners, Diverse Courses, Diverse Projects: Learning from Challenges in New Directions*, with Brook Osborne, Jeff Gray, Irene Lee, Bradley Beth, SIGCSE, Atlanta, GA, March 2014.
- A Public/Private Partnership for Expanding Computer Science in Schools*, with Amy Briggs, Brook Osborne, Pat Yongpradit, Gail Chapman, Joanna Goode, SIGCSE, Atlanta, GA, March 2014.
- Education for a Cyber-Capable Citizenry: From Majors to Nonmajors, From Ethics to Understanding*, with Brook Osborne, American Association of Colleges and Universities (AACU), Transforming STEM Education, October 2013, San Diego, CA
- CS Principles — Piloting a Portfolio Assessment*, CSTA Conference, Boston, MA, 2013, with Brook Osborne, Rich Kick, Rebecca Dovi, Baker Franke, Deepa Muralidhar
- A New AP Course: Computer Science Principles*, with Lien Diaz, Amy Briggs, Jill Kushner, Andrew Kuemmel, Deepa Muralidhar, AP Annual Conference (College Board), Las Vegas, NV, July 2013.
- Learning with Exploring Computer Science: Connections to AP Computer Science* with Lien Diaz, Amy Briggs, Gail Chapman, Joanna Goode, Jody Paul, AP Annual Conference (College Board), Las Vegas, NV, July 2013.
- CS Principles: Development and Evolution of a course and Community*, SIGCSE, Denver, CO, 2013, with Amy Briggs, Lien Diaz, Brook Osborne.
- Broadening Participation: ECS and CS Principles*, CS&IT, Irvine, CA, 2012, with Gail Chapman and Brook Osborne.
- CS Principles: A new Course in Computer Science*, North Central Business Education Association, 2012, Indianapolis, Indiana, with Brook Osborne.
- Update on the CS Principles Project*, SIGCSE, Raleigh, NC, 2012, with Amy Briggs, Jan Cuny, Lien Diaz, Chris Stephenson.

CS Principles: Piloting a National Course II, SIGCSE, Raleigh, NC, 2012, with Amy Briggs, Ralph Morelli, Dwight Barnet, Jeff Gray.

CS Principles: Piloting a New Course at National Scale, SIGCSE, Dallas, 2011 (with Larry Snyder, Tiffany Barnes, Dan Garcia, Jody Paul, Beth Simon).

The CS10K Project: Mobilizing the Community to Transform High School Computing SIGCSE, Dallas, 2011 (with Jan Cuny, Chris Stephenson, and Cameron Wilson)

The CS/10K Project, CRA/Snowbird Conference, Snowbird UTAH, July, 2010.

Code as a Metaphor for Computational Thinking, CSTA/CSIT Symposium, Google, Mountain View, CA, July, 2010.

Re-imagining the First Year of Computer Science, SIGCSE, Milwaukee, WI, 2010 (with Lien Diaz, Chris Stephenson, Jan Cuny, Amy Briggs)

FOSS Workshop, Free and Open Source Software, SIGCSE, Chattanooga, TN, 2009, invited speaker.

Computational Thinking Panel, SIGCSE, Chattanooga, TN, 2009 (with Amber Settle, Susanne Hambrusch, and Joan Peckham).

Advanced Placement Computer Science: The Future of Tracking the First Year of Instruction, Special Session, SIGCSE, Chattanooga, TN, 2009 (with Henry Walker, Chris Stephenson, Lien Diaz, and Jan Cuny)

Nifty Assignments, Special Session, SIGCSE, Chattanooga, TN, 2009 (with Nick Parlante)

Innovating our Self Image Special Session, SIGCSE, Portland, OR, 2008 (with Peter Denning)

Teaching Tips We Wish They Told Us Before We Started Special Session, SIGCSE, Cincinnati, OH, 2007 (with Dan Garcia, Nick Parlante, Stuart Reges).

Resolved: Objects Early Has Failed SIGCSE, St. Louis, 2005, Special Session (with Stuart Reges, Kim Bruce, Michael Kölling, Elliot Koffman).

But it Looks Rights: Bugs Students Don't See SIGCSE, Norfolk, 2004, Special Session (with David Ginat, Daniel Garcia, Mark Guzdial).

Colorful Illustrations of Algorithmic Design Techniques SIGCSE, Charlotte, 2001, Special Session (with David Ginat, Joseph Bergin, Dan Garcia).

Nifty Assignments in CS1 and CS2, Panelist, SIGCSE, Charlotte, 2001 (with Michael Clancy, Nick Parlante, Rich Pattis, Stuart Reges, Julie Zelenski).

FYI 2000: First Year Instruction, developed, organized, and chaired a workshop on first year instruction in computer science. The workshop had invited talks from both industry (Jon Bentley) and academia (Shriram Krishnamurthy, Richard Pattis) and was attended by more than 40 faculty from across the country. The workshop was sponsored by NSF, Microsoft, and the Duke Computer Science Department.

Patterns in Computer Science, (co-organizer with Eugene Wallingford), SIGCSE. Austin, TX. March 2000.

The Future of Advanced Placement Computer Science (panel). *SIGCSE Technical Symposium on Computer Science Education*. Austin, TX, 2000. With Corky Cartwright, David Gries, Cay Horstmann, Richard Kick, Fran Trees, Henry Walker, Ursula Wolz.

Nifty Assignments in CS1 and CS2, Panelist, SIGCSE, New Orleans, 1999 (with Michael Clancy, Nick Parlante, Rich Pattis, Stuart Reges, Julie Zelenski).

Incorporating Patterns into CS courses and Writing Patterns for CS Courses (co-organizer with Eugene Wallingford). SIGCSE, New Orleans, March 1999.

Future Directions in CS2 and Data Structures. Organized and ran the workshop that was held in conjunction with OOSPLA-98, Vancouver, CA, October 1998 (20 participants)

Object Oriented Design. Invited Participant, OOPSLA-97, Atlanta, Georgia, 1997.

Teaching Object-Oriented Programming: Practical Examples Using C++ and Java, Tutorial at PLDI 97, Las Vegas, Nevada, June 1997.

Future Directions in Data Structures and CS2. Organized and ran two-day workshop held at Duke co-sponsored by NSF, 32 participants, March, 2000.

Teaching Object-Oriented Design in the first year. Invited speaker and participant. OOPSLA-96, San Jose, CA, October, 1996.

Strategic Directions in Computing Research (SDCR), working group in Computer Science Education. Sponsored by ACM, CRA, and NSF, Boston, MA, June, 1996.

How to teach C++ in Introductory Courses, Tutorial part of PLDI, FCRC 1996, Philadelphia, PA, sponsored by SIGPLAN

Formal Methods Considered [Help — Harm]ful: Engaging students in the first year. Exploring Formal Methods in the Early Computer Science Curriculum, Joint NSF/US Department of Education Workshop. September 16, 1995 (invited speaker).

Developing an Object-Oriented Class Library. NSF sponsored workshop. Colgate University, June 1995. (invited participant)

A Sorcerer’s Apprentice Approach to using C++ in CS1. NECUSE (New England Consortium on Undergraduate Science Education) Workshop in Introductory Computer Science Curricula. January 1995.

Measuring Performance of Automated Theorem Provers (with D. W. Loveland). *Twelfth Conference on Automated Deduction (CADE-12)*, Workshop on Evaluation of Automated Theorem Proving Systems. Nancy, France, 1994.

Acquiring Object-Oriented Technology: A Bridge between Industry and Academia (invited participant). US West, Boulder Colorado, March 1994.

OOP: An introduction for Secondary School Teachers. Workshop delivered to secondary school computer science teachers in Dallas, TX, August 1993.

Simulation: A vehicle for exploring OOP. *Object-Oriented Curriculum Development Workshop*. NSF sponsored workshop. Colgate University, July 1993. (with Claire Bono)

A Tapestry of Fundamental Ideas and Concepts in Computer Science: a Programming, Contextual View for Liberal Arts Students *L**3: Logic, Loops, and Literacy*. NSF sponsored workshop on computer science for non-majors, Brooklyn College, May 1993.

Human Interaction with a High-Performance Theorem Prover. *International Joint Conference on Artificial Intelligence*. Workshop on Automated Theorem Proving. Chambéry, France, 1993. (with D.W. Loveland)

Logic Programming Considered Harmful? *Joint International Conference on Logic Programming*. Prolog as a first language track, Washington DC, November 1993.

Caching to reduce redundancy in Model Elimination Theorem Provers. *Joint Japanese-American workshop in Automated Theorem Proving*, Argonne National Labs, June 1991.

The METEOR implementations of the Model Elimination procedure. *Workshop on Proof Theory and Automated Theorem Proving*. Oberwolfach, Germany, April 1991. (with D.W. Loveland)

Online Exams in Advanced Placement Computer Science. *National Council of Teachers of Mathematics Conference*. San Francisco, April 1984.

VIII. Invited Lectures, Talks, and Workshops.

CS Principles: Present and Future, with Brook Osborne, NSF CE21/Delaware Project, June 2013.

CS Principles: Present and Future, with Brook Osborne, NSF CE21/Alabama Project, June 2013.

CS Principles: Present and Future, with Brook Osborne, NSF CE21/Mobile Project, Hartford CT, July 2013.

CS Principles: Present and Future, with Brook Osborne, NSF RETHink CS Project, Philadelphia, PA, July 2013.

Code: How to Use it, NSF RETHink CS Project, Philadelphia, PA, July 2013.

Code as a Metaphor for Computational Thinking, Distinguished Faculty Lecture, Virginia Tech, February 24, 2012

CS Principles, AP Annual Conference, July 2012, Orlando, FL.

CS Principles: Report and Progress, NCWIT National Summit, May, 2012.

CS Principles, AP Annual Conference, July 2011, San Francisco.

CS Principles: Report and Progress, NCWIT National Summit, May, 2011.

Developing a New, National Course in Computer Science, UNC Greensboro, January 25, 2011.

Code as a Metaphor for Computational Thinking, Harambeenet workshop, Durham, NC, July 2010.

CS Principles and the CS10K project, NSF, BPC Community Meeting, Los Angeles, February, 2010.

CCSC: Midwest, Keynote Speaker, "A New Way of Thinking about Computational Thinking", St. Xavier University, Chicago, October 2009.

National Academies: Workshop on Computational Thinking February 2009.

Problem-Centric Learning, Sept 2008, Rochester Institute of Technology

What is Computer Science?, April 2008, NSF CPATH, Living in the Knowledge Society.

Problems in AP Computer Science, June 2008, Advanced Placement Computer Science Reading, Professional Development Night.

CPATH: Science and Computer Science Purdue University, November 2007.

CPATH: Problem-based Learning Keynote, Workshop sponsored by Argonne Labs and Governors University, November 2007.

Microsoft Computational Thinking Summit Redmond, WA, September 2007.

Google Faculty Summit, Mountain View, CA, July 2007.

HarambeNet: Introducing Computer Science through Modeling and Analysis of Social Networks SIGCSE 2007 Workshop, with Jeffrey Forbes.

The Cruelty of Really Teaching Computer Science Redux, University of California, Riverside, January 2006.

The Cruelty of Really Teaching Computer Science Redux, University of British Columbia Computer Science Distinguished Lecturer Series, Fall 2005.

The Cruelty of Really Teaching Computer Science Redux, University of Washington Computer Science Distinguished Lecturer Series, Fall 2005.

The Cruelty of Really Teaching Computer Science Redux, Keynote talk at CCSC/SE, Consortium of Computing Sciences in Colleges, Southeast US, November 2005.

The Cruelty of Really Teaching Computer Science Redux, Keynote talk at CCSC/E, Consortium of Computing Sciences in Colleges, Eastern US, October 2005.

A Random Walk Through Computer Science, Invited/Keynote Talk at ACM/Student conference Reflections/Projections, University Illinois, Oct. 2004.

20 Years of Teaching Computer Science, invited talk at NSF Workshop for high school teachers, Stonehill College, October 2004.

Everything I Needed to Know about Programming and Computer Science I Learned from my Teachers, Keynote Talk, SIGCSE, Norfolk 2004.

Using Patterns in the First Year, invited tutorial and presentation as part of the 2000 Eastern Small College Computing Conference, University of Scranton, PA.

OO Design and Patterns, invited speaker at NSF-sponsored workshop for high school teachers at Stonehill College, June 2000.

Advanced OO Programming, invited speaker at NSF-sponsored workshop for high school teachers at Stonehill College, January 1999.

Object-Oriented Design and Programming. Three-day lecture/workshop co-taught with David Gries delivered to educators from business colleges in Denmark, October, 1998.

Possible Futures for CS2 (panelist). *SIGCSE Technical Symposium on Computer Science Education*. Atlanta, GA, 1998.

Teaching C++ in AP Courses: Four day workshop designed and delivered for the College Board, June and August, 1997.

The First Computer Science Course and C++: Paradigm Lost or Regained. DIMACS workshop on Discrete Mathematics, July, 1996.

C++ in the Advanced Placement Program. AP Computer Science Reading, Professional Night, Clemson, SC. June, 1996.

Use of C++ for CS1 and CS2, Computing Science Conference, Philadelphia, PA, 1996.

The First Year: Beyond Language Issues, (moderator and proposer), *SIGCSE Technical Symposium on Computer Science Education*. Philadelphia, PA, 1996.

Advanced Placement and C++: Opening a Dialogue, (moderator and proposer), *SIGCSE Technical Symposium on Computer Science Education*. Philadelphia, PA, 1996.

Object-Oriented Programming: How to “Scale Up” CS1. *SIGCSE Technical Symposium on Computer Science Education*. Phoenix, Arizona, 1994.

Themes and Tapestries: A Diversity of Approaches to Computer Science for Liberal Arts Students. *SIGCSE Technical Symposium on Computer Science Education*. Phoenix, Arizona, 1994.

Using Case Studies in Computer Science Courses. *SIGCSE Technical Symposium on Computer Science Education*. Phoenix, Arizona, 1994.

On Computer Science and Teaching Computer Science with some perspectives from automated Reasoning. Bryn Mawr College, January 1993.

Faster, Fairer and More Consistent Grading Techniques: Lessons From the Advanced Placement Reading *SIGCSE Technical Symposium on Computer Science Education*. Washington D.C., 1990.

The Pleasures and Perils of Teaching Introductory Computer Science. *North Carolina Council of Teachers of Mathematics Conference*, November 1990.

Teaching Recursion in Introductory Computer Science Courses. *North Carolina Council of Teachers of Mathematics Conference*. November 1986.

IX. Professional Affiliations

Member of ACM, IEEE Computer Society, SIGPLAN, SIGCSE, SIGACT, SIGCHI, SIGART, SIGSOFT, Sigma Xi.

X. Research Funding

2014 NSF CNS Collaborative Research: CS10K: Infusing Cooperative Learning into Computer Science Principles Courses to Promote Engagement and Diversity, \$104,714 (collaborative proposal with Rutgers and U. Alabama, this is Duke portion).

2014 NSF REU Supplement \$8,000 to support REU student Katharine Cummings, supplement to NSF 1246919

2014 NSF Supplement \$15,000 supplement to support travel for CS Principles, NSF grant 1246919

2013 NSF DRK-12 \$127,151 *Enhancing Computer Science Education in Grades 9-12 (ECSE)*, collaborative proposal to Duke, with Twin Cities Public TV, submitted, declined.

2013 NSF DRK-12 \$2.5 million *Enhancing Computer Science Education in Grades 90-12 (ECSE)*, main grant (PI) proposal to Twin Cities Public TV, submitted, declined.

2013 NSF REU Supplement \$7,680 to support CSURF student Eleanor Mehlenbacher.

2013, NSF STEP, *Addressing Retention at its Base: Building Strong Foundations in Early Courses through Mastery Learning*, \$107,021, submitted, *declined*.

2013, NSF CNS, *Collaborative Research: Broadening Participation in Computer Science: AP Computer Science Principles Phase II*, award to College Board, PI \$5,210,014

- 2013, NSF CNS, *Collaborative Research: Broadening Participation in Computer Science: AP Computer Science Principles Phase II*, award to Duke University, \$543,900
- 2011, NSF IIS, Special Projects: *Using Computational Thinking to Model a New Course: Advanced Placement Computer Science: Principles*, \$402,441, supplemental funding request to College Board, PI.
- 2009, NSF IIS, Special Projects: *Using Computational Thinking to Model a New Course: Advanced Placement Computer Science: Principles*, \$2,093,450.00, funding to College Board, PI.
- 2008, NSF BPC, Computational Thinking and Fluency in the 21st Century. \$98,415. Submitted by College Board, PI.
- 2007, NSF CPATH, CISE Distinguished Education Fellow *Interdisciplinary Problem- and Case-based Computer Science*, \$250K over two years, one of two inaugural CDEF awardees in Computer Science.
- 2004 IBM Faculty Fellow, *Issues in Large-scale Software Componentization*, \$40K grant.
- 2002-2003 \$28K, IBM Echelon: Eclipse Help for Learning Online.
- 2002 \$400K. IBM SUR, Education/Research Grant for establishing COD (Cluster On Demand) (co-PI with Richard Lucic, Amin Vahdat, Jeff Chase).
- 2002 \$1,500, OOPSLA Educator's Grant: Using Design Patterns (declined)
- 2001 \$1,500, OOPSLA Educator's Grant: Using Design Patterns (declined)
- 2001 \$120,000 IBM Education/Research Grant for establishing a teaching/cluster classroom (co-PI with Susan Rodger, Richard Lucic, Kishor Trivedi, Amin Vahdat, Jeff Chase, the \$120,000 is just the education part of the grant, each of three groups was awarded a similar amount. This was part of a \$1.7 million SUR grant from IBM, some of which was software and related support.)
- 2000-2005, \$480,826, NSF, *Modules and Courses for Ubiquitous and Mobile Computing*, NSF CRCO 0088078, PI, co-PIs Prof. Carla Ellis, Prof. Amin Vahdat.
- 2000, \$1.19 million Microsoft *Interactive Research/Teaching Classroom*, with Jeffrey Vitter, Richard Lucic, Jeffrey Chase, Carla Ellis, Deitolf Ramm, Susan Rodger, Amin Vahdat. This includes \$750,000 in software support and the rest in equipment, construction, and staffing support.
- 1998 \$223,179 equipment *Establishing Interactive Collaborative Classrooms* Hewlett-Packard University Grants Program, co-PI with Susan Rodger
- 1998 \$1,500, OOPSLA Educator's Grant: Using Design Patterns
- 1998 \$50,000 Microsoft Educational Development Grant, co-PI with Susan Rodger and Jeffrey Vitter
- 1998-2001 \$150,306 U.S. Dept of Education GANN (co-PI with Jeff Chase, Carla Ellis, Alvin Lebeck, Jeffrey Vitter)
- 1997-1998 \$80,000 equipment (part of a \$1.6 million grant) from Intel supporting computer science education at Duke.
- 1997 \$1,200, OOPSLA Educator's Grant: Using Design Patterns
- 1997-2002, \$200,004, "Using and Developing Design Patterns", National Science Foundation: CAREER Program, CAREER #9702550
- 1996, \$1,700, OOPSLA Educator's Grant: Using Design Patterns
- 1996-1997, \$119,382, "The Applied Apprenticeship Approach (AAA): An Object-Oriented/Object-Based Framework for CS2", National Science Foundation Course and Curriculum Development, grant#DUE-9554910.
- 1996-2001, \$607,800, "CURIOUS: The Center for Undergraduate Education and Research: Integration Through Performance and Visualization", NSF CISE Educational Infrastructure Program, grant #CISE-9634475 (co-PI with Prof. Susan Rodger)
- 1996-1998, \$13,080 "U.S.-Germany Cooperative Research to Enhance the Performance of the Model Elimination Proof Procedure" (co-PI with Prof. Donald Loveland), National Science Foundation INT-9514375

1995, \$1,000, OOPSLA Educator's Grant: Design Patterns in the Introductory CS Curriculum.

XI. Research Interests

Problem-centric Learning, Software Architecture, Object-oriented systems and languages, Computer Science Education, Networked and Distributed Computing, Automated Theorem Proving, Automated Reasoning, Parallel and Distributed Computing.

XII. Teaching

2002, Winner of the Richard K Lublin Award. Cited for "Ability to engender genuine intellectual excitement, ability to engender curiosity, knowledge of field and ability to communicate that knowledge, organizational skills, creative arrangement of course."

1999, Winner of Outstanding Instructor of Computer Science Award while on sabbatical at University of British Columbia (teaching CPSC 252, a course on Data Structures for approximately 250 Engineering students).

1995, Trinity College, Robert B. Cox Distinguished Teaching in Science Award. Cited for "knowledge of field and ability to communicate it to students, openness to students, skill in organizing courses, commitment to teaching over time."

1994, 1996, 2001, 2002 Nominated by undergraduates for outstanding teaching/faculty award (one of approximately 30 faculty nominated campus-wide each year).

Course and Curriculum Design/Implementation

CPS 53 – Program Analysis and Design I, Fall 1993

In 1993 I led the redesign of the core courses for majors (CPS 06, 08, and 100) to introduce object-oriented programming using C++ and an apprentice style of learning. This led to the development of new courses, described below. This redesign was a significant departure from the C-based courses that had been in place for three years requiring a large-scale change in philosophy as well as significant efforts in developing programming libraries to make the use of C++ feasible for students with no programming experience. This redesign has led to several publications and an NSF grant awarded in December of 1995.

CPS 100E - Program Analysis and Design II, Fall 1995 –

Created a new course for students with programming experience acquired elsewhere (not at Duke), replacing Computer Science 8 and accelerating students into the major. The course reviews material from the end of CPS 6 and then covers material from CPS 100. A formal laboratory component makes this possible. (with S. Rodger)

CPS 108 – Software Design and Implementation, Spring 1995 –

Designed and taught a new course required for all majors (in 1994 formalized the requirement). The course covers advanced object-oriented programming; introducing GUI programming using C++ and Java while emphasizing significant individual and team projects using object-oriented design, analysis, and programming.

In 1995 I established a new software design and engineering component of the curriculum via the course CPS 108. This curricular change led to an NSF CAREER grant for using and developing patterns in teaching software design and introductory programming.

CPS 149S - Problem Solving Seminar, Fall 1994

Created a new seminar course for problem solving, to prepare students for the ACM programming contest. Students worked previous contest problems once a week, and two mini-contests were held. Two teams participated in the regional contest with one team placing first.

CPS 182S – Technical and Social Analysis of Information and the Internet

Designed to meet the needs of Duke's Curriculum 2000. Satisfies, research and writing requirements and science technology and society requirement.

The development of technical and social standards governing the Internet and Information Technology in general. The role of software as it relates to law, patents, intellectual property, and IETF (Internet Engineering Task Force) standards. Written analysis of issues from a technical perspective with an emphasis on the role of software; but also on how standards relate to social and ethical issues.

In 2002 I designed and had approved CPS 182S, *Technical and Social Analysis of Information and the Internet* a course which R, W, and STS designations as part of Duke's Curriculum 2000 (research, writing, and science, technology and society, respectively). This course led to another, non-major's version of the course in 2008, to publications, and is part of the genesis for the new NSF CS Principles project.

CPS 004G – Programming for Bioinformatics

Designed as one course in a four-course, integrative and interdisciplinary program *The Genome Revolution: Society and Science* for first year students as part of Duke's FOCUS program. The course introduces programming in the context of solving problems from bioinformatics and computational biology.

In 2006 I used this course as a foundation, with work done by Alex Hartemink in our department on a more advanced course, to help spearhead and oversee the process leading to the approval of Duke's first interdepartmental and interdisciplinary minor: Computational Biology and Bioinformatics.

Compsci 82, Technical and Social Foundations of the Internet

In 2008 I used Compsci 182S (see above) as a model for developing Compsci 82, a course without the writing component, but with an Ethical Inquiry (EI) designation. This course was taught in the fall of 2008 to 239 students, the third largest enrollment for a one-section course at Duke. In 2009 I taught this course to 345 students, the second largest course at Duke and the largest course that does not satisfy a major requirement. In 2010 I again taught the course to 345 students; the course was the largest single-section course taught at Duke.

Compsci 6, Introduction to Computer Science

In 2010 I oversaw the development of a new, introductory course in computer science: Compsci 6, the new description for the course follows.

Introduction practices and principles of computer science and programming and their impact on and potential to change the world. Algorithmic, problem-solving, and programming techniques in domains such as art, data visualization, mathematics, natural and social sciences. Programming using high-level languages and design techniques emphasizing abstraction, encapsulation, and problem decomposition. Design, implementation, testing, and analysis of algorithms and programs. No previous programming experience required.

This course is intended to appeal to a wider and more diverse audience than our previous version of the course. I developed the infrastructure for the course including developing materials used to teach Python, deciding on the libraries used, and developing the software infrastructure to support the use of Python. I worked with Robert Duvall to deliver the first version of this course in the fall of 2010.

Compsci 92, Information and the Internet

In 2013 I piloted a new version of 82 (see above) with a strong quantitative component. The new course will be an exemplar for CS Principles. I have offered that course for three years, each spring in 2013-2015. It is now at a reasonable steady-state with excellent student evaluations, an integrated two-hour lab, both a data and a programming component, and work by students to understand ethical issues related to computing in science and society.

Courses Taught

In the list below, CPS 08/53 corresponds to CS 1 and CPS 100/103 corresponds to CS 2 (courses were renumbered in 1994). CPS 206 is a graduate level programming-languages course. CPS 10 is a comprehensive/breadth first introduction to Computer Science for non-majors.

Date		Number	Title	Enrollment
2015	Fall	CompSci 101	Introduction to Computer Science	201
2015	Spring	CompSci 92	Information and the Internet	92
		CompSci 342s	Technical and Social Foundations of the Internet	18
2014	Fall	<i>teaching leave</i>		
2014	Spring	CompSci 92	Information and the Internet	81
		CompSci 342s	Technical and Social Foundations of the Internet	18
2013	Fall	CompSci 110	Information, Society, the Internet (Bass)	42
2013	Spring	CompSci 92	Information and the Internet	71
		CompSci 342s	Technical and Social Foundations of the Internet	18
2012	Fall	CompSci 101	Introduction to Computer Science	225
		CompSci 201	(oversight course taught by others)	124
2012	Spring	CompSci 101	Introduction to Computer Science	143
		CompSci 182s	Technical and Social Foundations	18
2011	Fall	CompSci 89s	Robotics/Service Learning	12
		CompSci 100	Data Structure and Algorithms	165
		CompSci 82	Technical and Social Foundations of the Internet	346
2011	Spring	CompSci 149s	Problem Solving Seminar	8
		CompSci 006	Introduction to Computer Science	142
		CompSci 182s	Technical and Social Foundations of the Internet	18
2010	Spring	CompSci 149s	Problem Solving Seminar	12
		CompSci 100	Program Design and Analysis II	55
		CompSci 182s	Technical and Social Foundations of the Internet	18
	Fall	CompSci 149s	Problem Solving Seminar	8
		CompSci 006	Introduction to Computer Science	79
		CompSci 82	Technical and Social Foundations of the Internet	345
2009	Spring	CompSci 149s	Problem Solving Seminar	11
		CompSci 100	Program Design and Analysis II	29
		CompSci 182s	Technical and Social Foundations of the Internet	20
	Fall	CompSci 149s	Problem Solving Seminar	8
		CompSci 100	Program Design and Analysis II	41
		CompSci 82	Technical and Social Foundations of the Internet	345
2008	Spring	CompSci 149S	Problem Solving Seminar	7+
		CompSci 100	Program Design and Analysis II	33
		CompSci 82S	Technical and Social Foundations of the Internet	23
	Fall	CompSci 82	Technical and Social Foundations of the Internet 239	
		CompSci 100	Program Design and Analysis II	39
		CompSci 149s	Problem Solving Seminar	8+
2007	Spring	CompSci 149S	Problem Solving Seminar	8+
	Fall	CompSci 4G	Genomics Programming (FOCUS)	16
		CompSci 108	Software Design	44
		CompSci 149s	Problem Solving Seminar	5+
2006	Spring	CPS 182s	Technical and Social Analysis of the Internet	20
		CPS 100	Program Design and Analysis II	24
		CPS 149S	Problem Solving Seminar	4+
	Fall	CPS 004g	Introduction to Programming with Genomics (FOCUS)	11
		CPS 100	Program Design and Analysis II	36
		CPS 149S	Problem Solving Seminar	4+
2005	Fall	CPS 004G	Introduction to Programming with Genomics (FOCUS)	17
		CPS 108	Software Design and Implementation	32
		CPS 149S	Problem Solving Seminar	3
2005	Spring	CPS 182s	Technical and Social Analysis of the Internet	15
		CPS 149S	Problem Solving Seminar	4
2004	Fall	CPS 006G	Introduction to Programming with Genomics (FOCUS)	15
		CPS 100	Program Design and Analysis II	35
		CPS 149S	Problem Solving Seminar	7
2004	Spring	CPS 100	Program Design and Analysis II	35
		CPS 149S	Problem Solving Seminar	7
		CPS 108	Software Design and Implementation	45
2003	Fall	CPS 006X	Program Design and Analysis I (honors)	20

Date		Number	Title	Enrollment
2002	Fall	CPS 182s	Technical and Social Analysis of the Internet	43
		CPS 149S	Problem Solving Seminar	5
2002	Spring	CPS 100	Program Design and Analysis II	105
2001	Fall	CPS 108	Software Design and Implementation	64
		CPS 06	Program Design and Analysis I	105
		CPS 149S	Problem Solving Seminar	13
2001	Spring	CPS 108	Software Design and Implementation	124
		CPS 100	Program Design and Analysis II	114
		CPS 189S	CS Education Seminar	3
2000	Fall	CPS 100	Program Design and Analysis II	88
		CPS 149S	Problem Solving Seminar	14
		CPS 189S	CS Education Seminar	6
2000	Spring	CPS 100	Program Design and Analysis II	107
1999	Fall	CPS 108	Software Design and Implementation	115
		CPS 06	Program Design and Analysis I	173
1998	Fall	CS 252 (UBC)	Data Structures,CS2	163
1998	Spring	CPS 108	Software Design and Implementation	94
		CPS 196	Advanced Topics in OO Technology (seminar)	14
1997	Fall	CPS 100	Program Design and Analysis II	61
		CPS 108	Software Design and Implementation	63
		CPS 149S	Problem Solving Seminar	18
1997	Spring	CPS 100	Program Design and Analysis II	72
		CPS 108	Software Design and Implementation	65
1996	Fall	CPS 100E	Program Design and Analysis II	70
		CPS 108	Software Design and Implementation	58
		CPS 149S	Problem Solving Seminar	17
1996	Spring	CPS 100	Program Design and Analysis II	72
		CPS 108	Software Design and Implementation	40
		CPS 208	Software Design and Implementation	15

Date		Number	Title	Enrollment
1995	Fall	CPS 6	Intro. to Program Design/Analysis I (Team taught with S. Rodger)	121
		CPS 100	Program Design and Analysis II (Team taught with S. Rodger)	56
		CPS 100E	Program Design and Analysis II (Team taught with S. Rodger)	55
		CPS 149S	Problem Solving Seminar (Team taught with S. Rodger)	14
	Spring	CPS 108	Software Design and Implementation	38
		CPS 8	Intro. to Program/Design Analysis I	70
1994	Fall	CPS 8	Intro. to Program Design/Analysis I	63
		CPS 100	Program Design and Analysis II	45
	Spring	CPS 8	Intro. to Program Design/Analysis I	67
		CPS 100	Program Design and Analysis II	43
1993	Fall	CPS 8	Intro. to Program Design/Analysis I	39
		CPS 100	Program Design and Analysis II	29
	Spring	CPS 100	Program Design and Analysis II	48
		CPS 206	Programming Languages (graduate)	6
1992	Spring	CPS 10	Fundamentals of Computing	67
1991	Fall	CPS 10	Fundamentals of Computing	135
1987	Fall	CPS 51	Introduction to Programming	90
1987	Spring	CPS 51	Introduction to Programming	101
1986	Fall	CPS 51	Introduction to Programming	196

Thesis Advising

- 2004: Megan Murphy, *The Uses of Pair Programming in Introductory Computer Science Courses*, thesis for graduation with distinction.
- 2004: Megan Gessner, *Generation of Spanish Verb Conjugations*, thesis for graduation with distinction.
- 2002: Donald Onyango, *Comparison of Educational Tools*, Masters Thesis.
- 1998: Matthew Kotler, *An interactive CD-based Guide to Duke University*, undergraduate thesis for graduation with Highest Distinction.
- 1998: Eric Jewart, *Programming in CS 0*, undergraduate thesis for graduation with High Distinction.
- 1998: Tafawa Kesler, *GOODS: A Design and Class Building Tool*, undergraduate thesis for graduation with Distinction.
- 1997: Chih-ping Fu, *Towards a Java Bean Building and Using Environment*, Masters Thesis.

Independent Study

- Fall 2006, Spring 2007 *Computer and Mathematical Models of Insulin Pathways* Tiffany Chen, graduation with distinction, interdepartmental major, Computer Science and Biology (supervised from Biology by Fred Nijhout).
- Fall 2005, Industry/Academic software development with .NET technologies (with Glaxo-Smith-Kline)
- Spring 2005, Web-tools for Russian Vocabulary
- Spring 2004, Agile Methods and Programming for Spanish

Fall 2002, Patterns for Networked Games
Fall 2002, Online Grading
Fall 2001, Database-backed web sites: issues and solutions.
Fall 2000, A Framework for an online Calendar System supporting IETF standards
Spring 1998, Developing a CD-based guide to Duke, Advanced OO Design: A Class Browser
Fall 1997, Advanced Object Oriented Design, Interactive Web-based Journaling, Developing a CD-based guide to Duke
Spring 1997, Graphical Debugging
Spring 1997, Distance Learning using a Java Whiteboard
Fall 1996, On-line help by harvesting information with a GUI front end.
Fall 1996, Using C++ in High School Teaching.
Fall 1995, Graphics and Game programming for the Macintosh (6 students).
Summer 1995, Using C++ in High School Teaching
Spring 1995, A GUI/OO interface for air-quality modeling.
Spring 1995, Graphics Programming for the Macintosh.
Spring 1995, Object-Oriented Programming with Smalltalk.
Fall 1994, Implementing an Online Teacher Course Evaluation Book.
Spring 1994, An application-driven approach to foundations of computer science.
Summer 1994, The role of Computer Science for secondary school mathematics teachers.
Summer 1994, Computation Structures and Machine Organization.
Fall 1989, Graphical Display and Manipulation of Data Structures for the Macintosh.