

## Adrienne M. Decker

*adrienne@buffalo.edu*

Curriculum Vitae

June 1, 2010

Home:

Office: University at Buffalo

Department of Computer Science and Engineering

201 Bell Hall

Buffalo, NY 14260 – 2000

Phone: (716) 645-3184

---

### EDUCATION

2007 - Ph.D., Computer Science and Engineering, University at Buffalo (SUNY)

Dissertation: *How Students Measure Up: An Assessment Instrument for Introductory Computer Science*

Advisor: William J. Rapaport

2001 - M.S., Computer Science and Engineering, University at Buffalo (SUNY)

Master's Project: *ASPAS: A Solution for Providing Application Services*

Advisor: Bina Ramamurthy

2001 - B.S., Computer Science, *Summa Cum Laude*, University at Buffalo (SUNY)

### RESEARCH INTERESTS

Computing Education, Introductory Computer Science Curriculum, Objects-First Introductory Curriculum, Using Games to Motivate Introductory Computing Concepts, Tools for Supporting Computing Education, Discrete Mathematics Education and Curriculum, Assessment Issues in Introductory Computing

### PROFESSIONAL EXPERIENCE

8/2007 – present      **Teaching Assistant Professor**, Department of Computer Science and Engineering  
*University at Buffalo (Buffalo, NY)*

Courses Taught: Introduction to Computer Science for Majors I & II, Discrete Structures, Data Structures, Programming for Non-Majors I (Graduate), Programming for Non-Majors I (Undergraduate), Computers – A General Introduction

8/2002 – 7/2007      **Lecturer**, Department of Computer Science and Engineering  
*University at Buffalo (Buffalo, NY)*

Courses Taught: Introduction to Computer Science for Majors I & II, Discrete Structures, Programming Languages, Programming for Non-Majors I & II (Graduate), Programming for Non-Majors I & II (Undergraduate), Great Ideas in Computer Science

- 5/2001 - 8/2002      **Instructor**, Department of Computer Science and Engineering  
*University at Buffalo (Buffalo, NY)*  
 Courses Taught: Introduction to Computer Science for Majors I, Programming for Non-Majors (Graduate), Great Ideas in Computer Science
- 8/2001 - 12/2001    **Instructor**, Dept. of Mathematics/Physics/Computer & Information Sciences  
*Niagara County Community College (Sanborn, NY)*  
 Courses Taught: Introduction to Computer Systems

## **ADDITIONAL EXPERIENCE**

- 1/2000 - 5/2001      **Teaching Assistant & Summer Course Grader**  
*Department of Computer Science and Engineering*  
*University at Buffalo (Buffalo, NY)*  
 Selected as Head Teaching Assistant in the Introduction to Computer Science for Majors I course for Spring 2001 semester
- 6/1997-1/2000      **Program Director/Assistant Program Director/Youth Program Coordinator**  
 AMF Thruway Lanes  
 (Cheektowaga, NY)  
 --Managed office staff of 4-6 people who coordinated all programs for the center.  
 --Responsibilities included managing bowler information for center databases, promotions and advertising, scheduling of events, banquet and party planning, organization of center resources for special events.
- 6/1997-6/1999      **Yearbook Advisor**  
 Cheektowaga-Sloan Union Free School District  
 (Cheektowaga, NY)  
 --Provided instruction to students on yearbook staff about yearbook publication by hosting instructional workshops and individualized assistance in both yearbook production and software applications.  
 --Improved the production of the yearbook with introduction of new publication software, UltraVision, Adobe Photoshop, PageMaker, and Paint Shop Pro

## **AWARDS**

- 2007                    *Milton Plesur Award for Excellence in Teaching*, University at Buffalo
- 2005                    *Best Overall Paper*, Twenty-first Annual CCSC Eastern Conference  
 Decker, A., Haydanek S. and Egert C. "When Objects Collide: Abstractions over Common Physics Problems for Capstone Projects in CS1", *The Journal of Computing Sciences in Colleges*, **21**(2), pp. 12-18.
- 2003                    *Best Overall Paper*, Nineteenth Annual CCSC Eastern Conference  
 Decker, A., "A Tale of Two Paradigms", *The Journal of Computing Sciences in Colleges*, **19**(2), pp. 238-246.

2003

*Best Paper Finalist, Nineteenth Annual CCSC Eastern Conference*  
Decker, A., "A Tale of Two Paradigms", *The Journal of Computing Sciences in Colleges*, **19**(2), pp. 238-246.

## PUBLICATIONS

### BOOKS

1. Alphonse C. and **Decker A.** *Class Notes for CSE 115*, Self-published for use in course, 2006-2010.
2. **Decker A.** *Instructor's Manual with Solutions for Walter Savitch's Absolute Java 2<sup>nd</sup> Edition*, Addison-Wesley, Electronic Supplement, 2005.
3. **Decker A.** *Instructor's Manual with Solutions for Walter Savitch's Absolute Java*, Addison-Wesley, Electronic Supplement, 2004.

### JURIED/PEER REVIEWED PUBLICATIONS

1. Alphonse, C., Caspersen M., and **Decker A.** (2007) "Killer 'Killer Examples' for Design Patterns", *Proceedings of the 38<sup>th</sup> SIGCSE Technical Symposium on Computer Science Education*, Covington, KY. pp. 228 – 232.
2. **Decker A.**, Egert C., and Ventura, P. (2006) "Through the Looking Glass: Reflections on Using Undergraduate Teaching Assistants in CS1", *Proceedings of the 37<sup>th</sup> SIGCSE Technical Symposium on Computer Science Education*, Houston, TX. pp. 46 - 50.
3. **Decker, A.**, Haydanek, S. and Egert, C. (2005) "When Objects Collide: Abstractions over Common Physics Problems for Capstone Projects in CS1", *Journal of Computing Sciences in Colleges*, **21**(2), pp. 12-18.
4. Ventura, P., Egert, C., and **Decker A.** (2004) "Ancestor Worship in CS1: Reexamining the Introduction of Arrays", *2004 OOPSLA Educator's Symposium*, Vancouver, BC, pp. 68 - 72.
5. **Decker, A.** and Ventura, P. (2004) "Claim this Class for Computer Science: A Non-Mathematician's Discrete Structures Course", *Proceedings of the SIGCSE Technical Symposium on Computer Science Education*, Norfolk, VA, pp. 442 – 446.
6. **Decker, A.** (2004) "How Students Measure Up: Creation of an Assessment Tool for CS1", *SIGCSE 2004 Doctoral Consortium: held in conjunction with the SIGCSE 2004 Technical Symposium on Computer Science Education*, Norfolk, VA.  
<http://www.metrostate.edu/~fitzgesu/DC04/Decker.html>
7. **Decker, A.** (2003) "A Tale of Two Paradigms", *The Journal of Computing Sciences in Colleges*, **19**(2), pp. 238-246.
8. **Decker, A.** (2003) "I Want to be a Computer Scientist When I Grow Up: Evaluating the Skills Necessary for Computer Science", *SIGCSE 2003 Doctoral Consortium: held in conjunction with*

*the SIGCSE 2003 Technical Symposium on Computer Science Education, Reno, NV.*  
<http://www.radford.edu/~sigcse/DC03/participants/decker.html>.

### ***OTHER PUBLICATIONS***

1. **Decker, A.** "How Students Measure Up: An Assessment Instrument for Introductory Computer Science." Doctoral Dissertation, UMI AAT 3261966, DAI-A 68(5) [Nov 2007], Buffalo, NY, June 2007.

### **WORKSHOPS AND SPECIAL SESSIONS**

#### ***WORKSHOP ORGANIZATION COMMITTEE***

1. Skrien, D., **Decker, A.**, Caspersen, M., Börstler, J., and Alphonse, C. *Good Examples for Exposing Bad Practice: The Eighth "Killer Examples" Workshop*, Workshop: OOPSLA 2009, Orlando, Florida, October 25-29, 2009.
2. Caspersen, M., Börstler, J., **Decker, A.**, and Alphonse, C. *Worked Examples for Sound Object-Oriented Pedagogy: A "Killer Examples" Workshop*, Workshop: OOPSLA 2008, Nashville, Tennessee, October 19-23, 2008.
3. Alphonse, C., Börstler, J., Caspersen, M., **Decker, A.**, and Kölling, M. *Process in OO Pedagogy: A "Killer" Workshop*, Workshop: OOPSLA 2007, Montreal, Quebec, Canada, October 21-25, 2007.
4. Alphonse, C., Caspersen, M., **Decker, A.**, and Trask, B. *Fifth "Killer Examples" for Design Patterns Workshop*, Workshop: OOPSLA 2006, Portland, OR, October 22-26, 2006.
5. Alphonse, C., Caspersen, M., Wong, S. and **Decker, A.** *Fourth "Killer Examples" for Design Patterns and Objects First Workshop*, Workshop: OOPSLA 2005, San Diego, CA, October 16-20, 2005.

#### ***DOCTORAL CONSORTIUM PARTICIPANT***

1. **Decker, A.** (2004) "How Students Measure Up: Creation of an Assessment Tool for CS1", *SIGCSE 2004 Doctoral Consortium: held in conjunction with the SIGCSE 2004 Technical Symposium on Computer Science Education, Norfolk, VA.*  
<http://www.metrostate.edu/~fitzgesu/DC04/Decker.html>
2. **Decker, A.** (2003) "I Want to be a Computer Scientist When I Grow Up: Evaluating the Skills Necessary for Computer Science", *SIGCSE 2003 Doctoral Consortium: held in conjunction with the SIGCSE 2003 Technical Symposium on Computer Science Education, Reno, NV.*  
<http://www.radford.edu/~sigcse/DC03/participants/decker.html>.

## CONFERENCE PRESENTATIONS

1. Skrien, D., Alphonse, C., **Decker, A.**, Börstler, J., and Caspersen, M. (2009) “The Eighth “Killer Examples” Workshop: Good Examples for Exposing Bad Practice”, A poster presented at *OOPSLA 2009*, October 25-29, 2009, Orlando, Florida.
2. **Decker, A.**, and Alphonse, C. (2009) “Closing the Feedback Loop: Using Historical Data and Student Feedback to Guide Course Changes”, A poster presented at *SIGCSE 2009*, March 4-7, 2009, Chattanooga, Tennessee.
3. **Decker, A.**, Alphonse, C., Börstler, J., and Caspersen, M.E. (2008) “Worked Examples for Sound OO Pedagogy: The Seventh “Killer Examples” Workshop”, A poster presented at *OOPSLA 2008*, October 19-23, 2008, Nashville, Tennessee.
4. Wang, G., McSkimming, B., Marzec, Z., Gardner, J., **Decker, A.**, and Alphonse, C. (2007) “Green: A Flexible UML Class Diagramming Tool for Eclipse”, A demo presented at *OOPSLA 2007*, October 21-25, 2007, Montreal, Quebec, Canada.
5. Alphonse, C., Börstler, J., Caspersen, M.E., **Decker, A.**, and Kölling, M. (2007) “Process in OO Pedagogy: The Sixth “Killer Examples” Workshop”, A poster presented at *OOPSLA 2007*, October 21-25, 2007, Montreal, Quebec, Canada.
6. Wang, G., McSkimming, B., Marzec, Z., Gardner, J., **Decker A.**, and Alphonse, C. (2007) “Green: A Flexible UML Class Diagramming Tool for Eclipse”, A poster presented at *OOPSLA 2007*, October 21 – 25, 2007, Montreal, Quebec, Canada.
7. Wang, G., McSkimming, B., Marzec, Z., Gardner, J., **Decker A.**, and Alphonse, C. (2007) “Green: A Flexible UML Class Diagramming Tool for Eclipse”, A demo given *OOPSLA 2007*, October 21 – 25, 2007, Montreal, Quebec, Canada.
8. Gardner, J., McSkimming, B., Wang, G. - with Faculty Advisors Alphonse, C., and **Decker A.** (2007) “Green: A Software Design Tool Developed for Students by Students”, A poster presented at the *University at Buffalo’s Celebration of Academic Excellence*, April 19, 2007, Buffalo, NY.
9. Alphonse, C., Caspersen M., and **Decker A.** (2007) “Killer ‘Killer Examples’ for Design Patterns”, A paper presented at the *38<sup>th</sup> SIGCSE Technical Symposium on Computer Science Education*, March 9, 2007, Covington, KY.
10. Alphonse, C., Caspersen, M., **Decker, A.** and Trask, B. (2006) “Fifth Killer Examples for Design Patterns and Objects First Workshop Results”, A poster presented at *OOPSLA 2006*, October 22-26, 2006, Portland, OR.
11. **Decker, A.** and Alphonse, C. “Does CS1 Have to Be So Syntactical?” (2006) A poster presented at *OOPSLA 2006 Educator’s Symposium*, October 23, 2006, Portland, OR.
12. **Decker A.**, Egert C., and Ventura, P. (2006) “Through the Looking Glass: Reflections on Using Undergraduate Teaching Assistants in CS1”, A paper presented at the *37<sup>th</sup> SIGCSE Technical Symposium on Computer Science Education*, March 2, 2006, Houston, TX.

13. Alphonse, C., Caspersen, M., **Decker, A.**, Kosa, M., and Wong, S. (2006) “*Objects First, Design Patterns Second: Lessons Learned from the 'Killer Examples' for Design Patterns and Objects First Workshops*”, A poster presented at the 37<sup>th</sup> SIGCSE Technical Symposium on Computer Science Education, March 3, 2006, Houston, TX.
14. Alphonse, C., Caspersen, M., Wong, S. and **Decker, A.** (2005) “Fourth Killer Examples for Design Patterns and Objects First Workshop Results”, A poster presented at *OOPSLA 2005*, October 16-20, 2005, San Diego, CA.
15. **Decker, A.**, Haydanek, S. and Egert, C. (2005) “When Objects Collide: Abstractions over Common Physics Problems for Capstone Projects in CS1”, A paper presented at the *CCSC 21<sup>st</sup> Annual Eastern Conference*, October 14, 2005, Iona College, NY.
16. Ventura, P., Egert, C., and **Decker A.** (2004) “Ancestor Worship in CS1: Reexamining the Introduction of Arrays”, A paper presented at the *2004 OOPSLA Educator’s Symposium*, October 26, 2004, Vancouver, BC.
17. **Decker, A.** and Ventura, P. (2004) “Claim this Class for Computer Science: A Non-Mathematician’s Discrete Structures Course”, A paper presented at *the 35<sup>th</sup> SIGCSE Technical Symposium on Computer Science Education*, March 6, 2004, Norfolk, VA.
18. **Decker, A.** (2003) “A Tale of Two Paradigms”, A paper presented at the *CCSC 19<sup>th</sup> Annual Eastern Conference*, October 18, 2003, Montclair State University, NJ.

#### PANEL SESSION PARTICIPATION

1. Egert, C., Ventura, P., and **Decker, A.** “Putting the ‘Fun’ Back in Fundamentals: Using Games to Teach Object-Oriented Design Early”, Panel Session: Computer Gaming, *ASEE St. Lawrence Section Conference 2005*, Binghamton University, Binghamton, NY, April 8-9, 2005.

#### INVITED TALKS

1. “Teaching Java to Novices”, *Guest Speaker – SUNY Fredonia First Annual High School Programming Competition* (Chairperson: Karen Ehrlich), Fredonia, NY, December 19, 2003

#### SERVICE

##### NATIONAL/INTERNATIONAL CONFERENCE COMMITTEES

2009	OOPSLA Educator’s Symposium
2011	SIGCSE 2011 – Workshops Co-Chair

##### REGIONAL CONFERENCE COMMITTEES

2008	Western New York CSTA’s “Conversation about Computing”
2008	Western New York CSTA’s “College Computing Fair” – Program Chair
2009	Western New York CSTA’s “College Computing Fair” – Program Chair

**ELECTED OFFICES**

2008-present Western New York CSTA Secretary

**JOURNAL REVIEWS**

2009 Computer Science Education

**CONFERENCE REVIEWING**

2003 – 2010 SIGCSE Technical Symposium on Computer Science Education  
 2004 – 2007, CCSC Eastern Conference  
 2009, 2010  
 2009 – 2010 CCSC Northeastern Conference  
 2005 – 2007, Conference on Innovation and Technology in Computer Science Education  
 2009 - 2010 (ITiCSE)  
 2009 OOPSLA Educator’s Symposium

**WORKSHOP REVIEWING**

2009 Good Examples for Exposing Bad Practice: The Eighth “Killer Examples” Workshop  
 2008 Worked Examples for Sound Object-Oriented Pedagogy: A “Killer” Workshop  
 2007 Process in OO Pedagogy: A “Killer” Workshop, OOPSLA 2007.  
 2006 Fifth “Killer Examples” for Design Patterns Workshop, OOPSLA 2006.  
 2005 Fourth “Killer Examples” for Design Patterns and Objects First Workshop

**DEPARTMENTAL SERVICE***Advisor/Supervisor Roles*

9/2009 – present Supervisor, CSE 113 Undergraduate Teaching Assistants  
 1/2004 – present Supervisor, CSE 115 Undergraduate Teaching Assistants  
 1/2005 – present Advisor, Undergraduate Chapter of the ACM Student Club  
 1/2005 – 6/2007 Supervisor, CSE Undergraduate Lab Assistants  
 1/2006 – 9/2008 Supervisor, CSE Student Mentors  
 9/2009 – 10/2009 Supervisor, CSE 101 Undergraduate Teaching Assistant

*Committee Chairmanships (Hiring Committees)*

2010 CSE 111-116 Undergraduate Teaching Assistants  
 2009 CSE 115 Undergraduate Teaching Assistants  
 2008 CSE Student Mentors  
 2008 CSE 115 Undergraduate Teaching Assistants  
 2007 CSE Student Mentors  
 2007 CSE 115 Undergraduate Teaching Assistants  
 2006 CSE Student Mentors  
 2006 CSE 115 Undergraduate Teaching Assistants  
 2005 CSE 115 Undergraduate Teaching Assistants  
 2004 CSE 115 Undergraduate Teaching Assistants

*Committee Chairmanships (Other Committees)*

2004 CSE Promotional Video Committee  
 2002 – 2003 Discrete Structures Committee

*Committee Service*

2002 – present Undergraduate Affairs Committee (UGAC)  
 2004 – 2005 Computer Science Curriculum Committee

2005 – 2006          Computer Science BA Program Revision Committee

*Miscellaneous Service*

2008                  Marshall, SEAS Commencement  
2006                  University at Buffalo Faculty-to-Faculty Articulation Conference  
2006                  Engineering Discovery Day (CSE Presentation)  
2006, 2007, 2009    CSE Freshmen Orientation (Group Discussion)  
2006                  Reviewer, CSE Graduate Conference  
2006, 2007          University at Buffalo Preview Day (CSE Presentation)  
2005, 2006, 2007, 2009    University at Buffalo Open House (CSE Presentation)

***PUBLISHER-SOLICITED REVIEWS OF BOOKS, BOOK PROPOSALS, AND CHAPTERS***

1.    Koffman, E. and Wolfgang, P. (2010) *Data Structures: Design & Use in Java*, Second Edition, John Wiley & Sons Publishing, Two chapter review.
2.    Sanders, B. and Cumaranatunge, C. (2007) *ActionScript 3.0 Design Patterns*. O'Reilly Publishers, Technical Reviewer for entire manuscript.
3.    Gersting, J. (2007) *Mathematic Structures for Computer Science*, Sixth Edition. W.H. Freeman & Co. Publishers, Entire Book Reviewed by Chapter.
4.    Dale, N. and Weems. C. (2006) *Programming and Problem Solving with Java*, Second Edition. Jones & Barlett Publishers, Entire Book Reviewed by Chapter.
5.    Koffman, E. *Objects, Abstraction, Data Structures and Design: Using Java*. John Wiley & Sons Publishing, Single chapter review.
6.    Carrano, F. and Savitch, W. (2005) *Data Structures and Abstractions with Java*. Prentice Hall Publishers, Entire Book Reviewed by Chapter.
7.    Hahn, H. *Harley Hahn's Student Guide to Unix*, Third Edition. McGraw-Hill Publishing, Potential Audience/TOC Review.
8.    Henry, M. *Discrete Math for Students*. McGraw-Hill Publishing, Recommendations on Book Proposal under consideration for publication.
9.    Savitch W. (2003) *Absolute Java*. Addison-Wesley, Book/Chapter Reviews.
10.    Wu, C.T. (2003) *An Introduction to Object-Oriented Programming with Java (3<sup>rd</sup> Edition)*. McGraw-Hill, Book/Chapter Reviews.
11.    Cohoon, J. and Davidson, J. (2004) *Java Program Design*, McGraw-Hill, Book/Chapter Reviews.
12.    Kamin, S., Mickunas, M.D., and Reingold E. (2002) *Pan Introduction to Computer Science Using Java (2<sup>nd</sup> Edition)*, McGraw-Hill, Book/Chapter Reviews



## TEACHING

### Courses Taught at University at Buffalo

<i>Session</i>	<i>Course</i> ** Course with multiple sections and multiple instructors. TAs would be split among all instructors for course.	<i>Class Size</i>	<i>Teaching Asst.</i> <i>U = Undergraduates G = Graduate</i>
Spring 2010	Introduction to Computer Programming I (CSE 113 Sections A & B)	86 + 111 = 197	1 ( <i>U – Teaching</i> ) 2 ( <i>G – Teaching &amp; Grading</i> )
Spring 2010	Introduction to Computer Science for Majors I (CSE 115 Section A)	115	3 ( <i>U – Teaching</i> ) 1 ( <i>G – Grading</i> )
Spring 2010	Computer Science for Non-Majors I (CSE 503)	12	Same as above
Fall 2009	Computers: A General Introduction (CSE 101)	109	3 ( <i>U – Teaching</i> ) 1 ( <i>G – Grading</i> )
Fall 2009	Introduction to Computer Programming I (CSE 113 Sections A & B)	70 + 121 = 191	1 ( <i>U – Teaching</i> ) 3 ( <i>G – Teaching &amp; Grading</i> )
Spring 2009	Introduction to Computer Programming I (CSE 113 Sections A & B)	85 + 121 = 206	3 ( <i>G</i> )
Spring 2009	Data Structures (CSE 250)	67	2 ( <i>G</i> )
Fall 2008	Introduction to Computer Science for Majors I (CSE 115 Sections A & B)**	72 + 45 = 117	3 ( <i>U – Teaching</i> ) 1 ( <i>G – Grading</i> )
Fall 2008	Data Structures (CSE 250)	58	2 ( <i>G</i> )
Spring 2008	Introduction to Computer Science for Majors I (CSE 115)	62	3 ( <i>U – Teaching</i> ) 1 ( <i>G – Grading</i> )
Spring 2008	Computer Science for Non-Majors I (CSE 503)	7	Same as above
Spring 2008	Data Structures (CSE 250)	62	2 ( <i>G</i> )
Spring 2008	Introduction to Computer Programming I (CSE 113 Sections A & B)	80 + 98 = 178	4 ( <i>G</i> )
Fall 2007	Introduction to Computer Science for Majors II (CSE 116)	31	2 ( <i>G</i> )
Fall 2007	Introduction to Discrete Structures (CSE 191)	80	3 ( <i>G</i> )
Spring 2007	Introduction to Computer Science for Majors I (CSE 115 Sections A & B)	46 + 43 = 89	3 ( <i>U – Teaching</i> ) 2 ( <i>G – Grading</i> )
Spring 2007	Computer Science for Non-Majors I (CSE 503)	6	Same as above
Spring 2007	Introduction to Discrete Structures (CSE 191)	77	2 ( <i>G</i> )
Fall 2006	Introduction to Computer Science for Majors I (CSE 115 Section C) [co-taught with another instructor]**	61	4 ( <i>U–Teaching</i> ) 4 ( <i>G–Grading</i> )
Fall 2006	Introduction to Computer Science for Majors II (CSE 116)	39	2 ( <i>G</i> )
Fall 2006	Computer Science for Non-Majors II (CSE 504)	1	Same as above
Fall 2006	Introduction to Discrete Structures (CSE 191 A & B)	35 + 45 = 80	3 ( <i>G</i> )
Spring 2006	Introduction to Computer Science for Majors I (CSE 115 Sections A & B)	62 + 65 = 127	3 ( <i>U–Teaching</i> ) 2 ( <i>G–Grading</i> )
Spring 2006	Computer Science for Non-Majors I (CSE 503)	9	Same as above
Spring 2006	Introduction to Discrete Structures (CSE 191)	67	2 ( <i>G</i> )
Fall 2005	Introduction to Computer Science for Majors I ** (CSE 115 Section C)	49	4 ( <i>U–Teaching</i> ) 2 ( <i>G–Grading</i> )

Fall 2005	Introduction to Discrete Structures (CSE 191)	38 + 51 = 89	3 (G)
Spring 2005	Introduction to Computer Science for Majors I (CSE 115 Sections A & B)	68 + 44 = 112	5 (U-Teaching) 1 (G-Grading)
Spring 2005	Computer Science for Non-Majors I (CSE 503)	2	Same as above
Spring 2005	Introduction to Discrete Structures (CSE 191)	65	2 (G)
Fall 2004	Introduction to Computer Science for Majors I ** (CSE 115 Section D)	48	5 (U-Teaching) 3 (G-Grading)
Fall 2004	Introduction to Discrete Structures (CSE 191 Section A & B)	118	3 (G)
Spring 2004	Introduction to Computer Science for Majors I (CSE 115 Sections A & B)	74 + 54 = 128	4 (U-Teaching) 2 (G-Grading)
Spring 2004	Computer Science for Non-Majors I (CSE 503)	14	Same as above
Spring 2004	Introduction to Discrete Mathematics (CSE 191)	55	1 (G)
Fall 2003	Introduction to Computer Science for Majors II (CSE 116 Sections A & B)	45 + 33 = 78	2 (G)
Fall 2003	Computer Science for Non-Majors II (CSE 504)	1	Same as above
Fall 2003	Introduction to Computer Programming I (CSE 113 Section B)	68	2 (G)
Spring 2003	Introduction to Discrete Mathematics (CSE 191)	73	2 (G)
Spring 2003	Introduction to Computer Programming II (CSE 114 Sections A & B)	46 + 67 = 113	3 (G)
Fall 2002	Introduction to Computer Science for Majors I ** (CSE 115 Sections C & D)	75 + 80 = 155	5 (U - Teaching ) 1 (G - Teaching) 2 (G - Grading only)
Fall 2002	Computer Science for Non-Majors I (CSE 503)	18	Same as above
Fall 2002	Great Ideas in Computer Science I (CSE 111)	183	4 (G)
Summer 2002	Programming Languages (CSE 305)	38	0
Spring 2002	Great Ideas in Computer Science I (CSE 111)	89	2 (G)
Spring 2002	Computer Science for Non-Majors II (CSE 504)	8	1 (G)
Fall 2001	Introduction to Computer Science for Majors I ** (CSE 115 Section A)	96	10 (G)
Summer 2001	Introduction to Computer Science for Majors I (CSE 115)	31	0

### Courses Taught at Niagara County Community College

<i>Session</i>	<i>Course</i>	<i>Class Size</i>	<i>Teaching Asst.</i>
Fall 2001	Introduction to Computer Systems	18	0

### Teaching Assistantships and Course Grading Assignments at University at Buffalo

<i>Session</i>	<i>Course</i>	<i>Class Size (approx.)</i>	<i>Course Type</i>
Spring 2001 (Head TA)	Introduction to Computer Science for Majors I (CSE 115)	100	Undergraduate
Fall 2000 (TA)	Introduction to Computer Programming II (CSE 114)	100	Undergraduate
Summer 2000	Introduction to Computer Literacy	25	Undergraduate

(Lab Assist.)	(CSE 101)		
Summer 2000 (Grader)	Introduction to Computer Science for Majors II (CSE 116)	44	Undergraduate
Summer 2000 (Grader)	Introduction to Computer Science for Majors I (CSE 115)	43	Undergraduate
Summer 2000 (Grader)	Introduction to Computer Programming I (CSE 113)	27	Undergraduate
Spring 2000 (TA)	Introduction to Computer Programming I (CSE 113)	100	Undergraduate

## PROFESSIONAL DEVELOPMENT

2010 – Attended SIGCSE 2010 Conference.

2009 – Attended Creating a Network: Connecting K through 16 Computing Education in WNY Conference

2009 – BlueJ/Greenfoot Day 2009 (Half-day mini-conference held at SIGCSE 2009)

2008 – Attended OOPSLA Educator’s Symposium [Educator’s Symposium Scholarship Recipient]

2008 – Writing More Effective NSF Proposals: Stephen Cooper, Timothy Fossum– Presenters (Workshop held at SIGCSE 2008)

2008 – Teaching and Testing the Middle Novice Programmer: Raymond Lister – Presenter (Workshop held at SIGCSE 2008)

2008 – BlueJ/Greenfoot Day 2008 (Half-day mini-conference held at SIGCSE 2008)

2007 – Green Bar for C++ - Unit Testing and Refactoring C++: Peter Somerlad – Presenter (Tutorial held at OOPSLA 2007)

2007 – Attended OOPSLA Educator’s Symposium [Educator’s Symposium Scholarship Recipient]

2007 – Teaching Objects First in an Enlightening, Exciting Manner: David Gries - Presenter (Workshop held at SIGCSE 2007).

2006 – Attended OOPSLA Educator’s Symposium [Educator’s Symposium Scholarship Recipient]

2005 – Attended OOPSLA Educator’s Symposium [Educator’s Symposium Scholarship Recipient]

2004 – Attended OOPSLA Educator’s Symposium [Educator’s Symposium Scholarship Recipient]

2004 – Designing with Patterns: John Vlissides – Presenter (Tutorial held at OOPSLA 2004).

2004 – Java Generics: Angelika Langer – Presenter (Tutorial held at OOPSLA 2004).

2004 – DMS 612 – Programming for Web Design: Christopher Egert - Instructor (Semester Course).

2002 – Attended SIGCSE 2002 Conference.

## SUPERVISED STUDENT RESEARCH PROJECTS

- 2006 - present     **Green UML Tool Development**  
*Students [Current]: John Kirchgraber, Daniel Padgett, Gene Wang, Sean Weppner*  
*Students [Former/Graduated]: Robert Dygert, Joshua Gardner, Zachary Marzec, Brian McSkimming, Austin Miller, Moses Vaughan*  
Supervision and guidance of the students providing the continual development of the Green UML Tool. Currently implemented as a plug-in to the Eclipse development environment, this tool is an open-source UML diagramming tool used by students in UB's CSE 115 and CSE 116 course. It is also available for download and is being used outside the university.
- 2008                **Traffic Controlling System Design & Study**  
*Student: Jianqiao Zhu (Undergraduate)*  
Research into methods of studying traffic flow and how to better control traffic patterns and traffic light synchronization by processing GPS information and using aspects of graph theory.
- 2007                **Undergraduate Research in Computer Science Education**  
*Student: Kari Bancroft (Undergraduate)*  
Initial development of a study to see what effect visualizations have on the learning experience for students in undergraduate computer science courses.
- 2007                **Game Design and Implementation: NES Platform**  
*Student: Mark Zorn (Graduate)*  
Exploration of the technologies used in the original 8-bit Nintendo gaming system. Student developed an understanding of the language, compiler, and limitations of this early gaming technology while creating a game in this environment.
- 2006                **CSE 115 Graphics Package**  
*Student: Michael Kozelsky (Graduate)*  
Development of graphics package for use by the students enrolled in CSE 115 (CS1) in the Java programming language. Package was built upon existing Java graphics capabilities, but designed to be easier for students to use so that the students could create graphical programs from scratch faster than with traditional raw Java code. Package has been used to facilitate creation of simulation environments and games by the students in the CS1 course. Package still used in course to present date.
- 2006                **Game Design and Implementation: Microsoft Platform**  
*Student: Jason Abofsky (Undergraduate)*  
Using Visual Studio, C++, DirectX and some basic graphics and gaming algorithms, a prototype game was developed as a sequel to work previously done by the student. This adventure-style game takes its main character Dabu through various worlds where he is required to pick up items and face off against menacing villains.

- 2006                    **Survey of Projects for Discrete Structures**  
*Student: Benjamin Robboy (Undergraduate)*  
 Creation of survey of published projects and activities for Discrete Structures courses created by student using published resources and websites. A prototype project was constructed to allow students to better explore the applications of the material in Discrete Structures to the computing field. Project assigned to students in Fall 2006 semester.
- 2006                    **CSE 115 Course Materials for Instructors and Teaching Assistants**  
*Students: W. Clark Dever, Michael Kozelsky, Jimmie Perrin (Undergraduates)*  
 Creation of reference materials for CSE 115 course including complete set of lecture notes, guidelines for teaching assistants with sample lesson plans and exploration of web technologies for improving course web site.
- 2004 - 2005           **CSE 115 Physics Package**  
*Student: Sara Haydanek (Undergraduate)*  
 Creation of a Physics API in Java that is integrated with NGP (Graphics Package used in CSE 115) and provides a framework for students for doing basic collision detection, gravity, and friction within their programs for CSE 115. Used for the first time in Spring 2005 for the final lab of CSE 115. Paper submitted and accepted to CCSC Eastern Conference 2005 and won Best Paper Award at conference.

## PERSONAL RESEARCH PROJECTS

- 2005                    **An API for APV**  
*Additional Project Completed while pursuing PhD*  
 Design for an object-oriented API for active and passive vision tasks. The main work of the project was an architectural document that outlined the various components (objects) needed for an API to support basic computer vision tasks in both 2D and 3D environments.
- 2000 - 2001           **ASPAS: A Solution to Providing Application Services**  
*Masters Project*  
 Designed and implemented an application server framework for use by an application service provider. Created a cross-platform model that allows application services to be written as either COM or EJB components to communicate with several heterogeneous clients using SOAP. A Java user interface was written for user interaction with the service.

## TECHNICAL SKILLS

<b>Operating Systems:</b>	Windows (up to Vista), Linux, Unix, MS-DOS
<b>Productivity Software:</b>	Microsoft Office, Adobe Dreamweaver, Open Office, Adobe Fireworks, Adobe Photoshop, Google Docs
<b>Programming Languages:</b>	Java, C++, C, ML, Ada, Prolog, Lisp, Visual Basic, Assembly
<b>Programming Environments:</b>	Eclipse, Emacs, Greenfoot, Dr. Java, BlueJ, Visual Studio
<b>Web Technologies:</b>	HTML, CSS, Javascript, PHP
<b>Databases:</b>	Oracle, MySQL

## **PROFESSIONAL AFFILIATIONS**

Association for Computing Machinery (ACM) and its special interest group:

SIGCSE – Computer Science Education

SIGITE – Information Technology Education

SIGPLAN – Programming Languages

Consortium for Computing Sciences in Colleges

Computer Science Teachers Association (CSTA)

Western New York CSTA Chapter:

Founding Member

Secretary 2008-present