Thomas Helmuth

198 College Hill Rd., Clinton, NY 13323
(315) 859-4507 • thelmuth@hamilton.edu
www.cs.hamilton.edu/~thelmuth/

Education

University of Massachusetts Amherst **Amherst, MA**
Ph.D., Computer Science *July 2015*
Thesis: General Program Synthesis from Examples Using Genetic Programming with Parent Selection Based on Random Lexicographic Orderings of Test Cases

University of Massachusetts Amherst **Amherst, MA**
M.S., Computer Science *May 2012*

Hamilton College **Clinton, NY**
Bachelor of Arts, summa cum laude *May 2009*
Computer Science with departmental honors; Mathematics

Appointments

Hamilton College **Clinton, NY**
Associate Professor of Computer Science *July 2023–present*
Assistant Professor of Computer Science *July 2017–June 2023*

Washington and Lee University **Lexington, VA**
Assistant Professor of Computer Science *July 2015–June 2017*

Research Interests

Genetic programming, evolutionary computation, automatic program synthesis

Awards and Memberhips

Class of 1963 Excellence in Teaching Award *2021*
Sigma Xi *2019-2024*
Association for Computing Machinery (ACM) *2012-present*

Publications

Peer-Reviewed Journal Articles

[1] Ryan Boldi, Martin Briesch, Dominik Sobania, Alexander Lalejini, Thomas Helmuth, Franz Rothlauf, Charles Ofria, and Lee Spector. Informed Down-Sampled Lexicase Selection: Identifying productive training cases for efficient problem solving. *Evolutionary Computation*, pages 1–31, March 2024.

[2] Thomas Helmuth and Peter Kelly. Applying genetic programming to PSB2: the next generation

program synthesis benchmark suite. *Genetic Programming and Evolvable Machines*, 23(3):375–404, September 2022. Special Issue: Highlights of Genetic Programming 2021 Events.

[3] Thomas Helmuth and Lee Spector. Problem-Solving Benefits of Down-Sampled Lexicase Selection. *Artificial Life*, pages 1–21, Sept 2021.

[4] Thomas Helmuth, Edward Pantridge, and Lee Spector. On the importance of specialists for lexicase selection. *Genetic Programming and Evolvable Machines*, 21(3):349–373, September 2020. Special Issue: Highlights of Genetic Programming 2019 Events.

[5] William La Cava, Thomas Helmuth, Lee Spector, and Jason H. Moore. A probabilistic and multi-objective analysis of lexicase selection and epsilon-lexicase selection. *Evolutionary Computation*, 27(3):377–402, Fall 2019.

[6] Thomas Helmuth, Lee Spector, and James Matheson. Solving uncompromising problems with lexicase selection. *IEEE Transactions on Evolutionary Computation*, 19(5):630–643, October 2015.

Peer-Reviewed Conference Publications

[7] Thomas Helmuth, Jayden Fedoroff, Edward Pantridge, and Lee Spector. Facilitating function application in code building genetic programming. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO '24, page 887–895, Melbourne, Australia, 2024. Association for Computing Machinery.

[8] Thomas Helmuth, Edward Pantridge, James Gunder Frazier, and Lee Spector. Generational computation reduction in informal counterexample-driven genetic programming. In Mario Giacobini, Bing Xue, and Luca Manzoni, editors, *EuroGP 2024: Proceedings of the 27th European Conference on Genetic Programming*, volume 14631 of *LNCS*, pages 21–37, Aberystwyth, Wales, 3-5 April 2024. Springer.

[9] Ryan Boldi, Ashley Bao, Martin Briesch, Thomas Helmuth, Dominik Sobania, Lee Spector, and Alexander Lalejini. Untangling the effects of down-sampling and selection in genetic programming. In *Artificial Life Conference Proceedings*, Copenhagen, Denmark, 22-26 July 2024. MIT Press.

[10] Edward Pantridge and Thomas Helmuth. Solving novel program synthesis problems with genetic programming using parametric polymorphism. In *Proceedings of the 2023 Genetic and Evolutionary Computation Conference*, GECCO '23, pages 1175–1183, Lisbon, Portugal, 15-19 July 2023. Association for Computing Machinery.

[11] Edward Pantridge, Thomas Helmuth, and Lee Spector. Functional code building genetic programming. In *Proceedings of the 2022 Genetic and Evolutionary Computation Conference*, GECCO '22, pages 1000–1008, Boston, USA, 9-13 July 2022. Association for Computing Machinery.

[12] Thomas Helmuth, Johannes Lengler, and William La Cava. Population diversity leads to short running times of lexicase selection. In Günter Rudolph, Anna V. Kononova, Hernán Aguirre,

Pascal Kerschke, Gabriela Ochoa, and Tea Tušar, editors, *Parallel Problem Solving from Nature – PPSN XVII*, pages 485–498. Springer International Publishing, August 15 2022.

[13] Thomas Helmuth and Peter Kelly. PSB2: The second program synthesis benchmark suite. In *Proceedings of the 2021 Genetic and Evolutionary Computation Conference*, GECCO '21, pages 785–794, internet, July 10-14 2021. Association for Computing Machinery. **Nominated, Best Paper Award, Genetic Programming Track**.

[14] Thomas Helmuth and Lee Spector. Explaining and exploiting the advantages of down-sampled lexicase selection. In *Artificial Life Conference Proceedings*, pages 341–349, Online, 13-18 July 2020. MIT Press.

[15] Thomas Helmuth, Edward Pantridge, Grace Woolson, and Lee Spector. Genetic source sensitivity and transfer learning in genetic programming. In *Artificial Life Conference Proceedings*, pages 303–311, Online, 13-18 July 2020. MIT Press.

[16] Thomas Helmuth, Edward Pantridge, and Lee Spector. Lexicase selection of specialists. In *GECCO '19: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1030–1038, Prague, Czech Republic, 13-17 July 2019. ACM. **Winner of Best Paper Award, Genetic Programming Track**.

[17] Lia Jundt and Thomas Helmuth. Comparing and combining lexicase selection and novelty search. In *GECCO '19: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1047–1055, Prague, Czech Republic, 13-17 July 2019. ACM.

[18] Thomas Helmuth, Nicholas Freitag McPhee, and Lee Spector. Program synthesis using uniform mutation by addition and deletion. In *GECCO '18: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1127–1134, Kyoto, Japan, 15-19 July 2018. ACM. **Winner of Best Paper Award, Genetic Programming Track**.

[19] Thomas Helmuth, Nicholas Freitag McPhee, Edward Pantridge, and Lee Spector. Improving generalization of evolved programs through automatic simplification. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO '17, pages 937–944, Berlin, Germany, 15-19 July 2017. ACM. **Nominated, Best Paper Award, Genetic Programming Track**.

[20] Thomas Helmuth, Nicholas Freitag McPhee, and Lee Spector. The impact of hyperselection on lexicase selection. In Tobias Friedrich, editor, *GECCO '16: Proceedings of the 2016 conference on Genetic and Evolutionary Computation Conference*, pages 717–724, Denver, USA, 20-24 July 2016. ACM. **Nominated, Best Paper Award, Genetic Programming Track**.

[21] Thomas Helmuth and Lee Spector. General program synthesis benchmark suite. In *GECCO '15: Proceedings of the 2015 conference on Genetic and Evolutionary Computation Conference*, pages 1039–1046, Madrid, Spain, 11-15 July 2015. ACM.

[22] William La Cava, Thomas Helmuth, Lee Spector, and Kourosh Danai. Genetic programming with epigenetic local search. In *GECCO '15: Proceedings of the 2015 conference on Genetic and Evolutionary Computation Conference*, pages 1055–1062, Madrid, Spain, 11-15 July 2015. ACM. **Nominated, Best Paper Award, Genetic Programming Track**.

[23] Thomas Helmuth and Lee Spector. Word count as a traditional programming benchmark problem for genetic programming. In *GECCO '14: Proceedings of the 2014 conference on Genetic and evolutionary computation*, pages 919–926, Vancouver, BC, Canada, 12-16 July 2014. ACM.

[24] Lee Spector, Kyle Harrington, and Thomas Helmuth. Tag-based modularity in tree-based genetic programming. In *GECCO '12: Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference*, pages 815–822, Philadelphia, Pennsylvania, USA, 7-11 July 2012. ACM.

[25] Lee Spector, Brian Martin, Kyle Harrington, and Thomas Helmuth. Tag-based modules in genetic programming. In *GECCO '11: Proceedings of the 13th annual conference on Genetic and evolutionary computation*, pages 1419–1426, Dublin, Ireland, 12-16 July 2011. ACM.

Refereed Workshop Publications

[26] James Gunder Frazier and Thomas Helmuth. Explaining automatically designed software defined perimeters with a two phase evolutionary computation system. In *Proceedings of the 2024 Genetic and Evolutionary Computation Conference Companion*, GECCO '24, Melbourne, Australia, 2024. Association for Computing Machinery.

[27] Thomas Helmuth, James Gunder Frazier, Yuhan Shi, and Ahmed Farghali Abdelrehim. Human-driven genetic programming for program synthesis: A prototype. In *Proceedings of the 2023 Genetic and Evolutionary Computation Conference Companion*, GECCO '23, page 1981–1989, Lisbon, Portugal, 2023. Association for Computing Machinery.

[28] Ryan Boldi, Ashley Bao, Martin Briesch, Thomas Helmuth, Dominik Sobania, Lee Spector, and Alexander Lalejini. The problem solving benefits of down-sampling vary by selection scheme. In *Proceedings of the 2023 Genetic and Evolutionary Computation Conference Companion*, GECCO '23, page 527–530, Lisbon, Portugal, 2023. Association for Computing Machinery.

[29] Ryan Boldi, Thomas Helmuth, and Lee Spector. The environmental discontinuity hypothesis for down-sampled lexicase selection. In *Artificial Life Conference Proceedings: Why It Didn't Work-Shop*, Online, July 18-22 2022. MIT Press.

[30] Li Ding, Ryan Boldi, Thomas Helmuth, and Lee Spector. Lexicase selection at scale. In *Proceedings of the 2022 Genetic and Evolutionary Computation Conference Companion*, GECCO '22, Boston, USA, July 9-13 2022. ACM.

[31] Hammad Ahmad and Thomas Helmuth. A comparison of semantic-based initialization methods for genetic programming. In *GECCO '18: Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pages 1878–1881, Kyoto, Japan, 15-19 July 2018. ACM.

[32] Edward Pantridge, Thomas Helmuth, Nicholas Freitag McPhee, and Lee Spector. Specialization and elitism in lexicase and tournament selection. In *GECCO '18: Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pages 1914–1917, Kyoto, Japan, 15-19 July 2018. ACM.

[33] Edward Pantridge, Thomas Helmuth, Nicholas Freitag McPhee, and Lee Spector. On the difficulty of benchmarking inductive program synthesis methods. In *Proceedings of the Genetic*

and Evolutionary Computation Conference Companion, GECCO '17, pages 1589–1596, Berlin, Germany, 15-19 July 2017. ACM.

[34] Thomas Helmuth, Nicholas Freitag McPhee, and Lee Spector. Effects of lexicase and tournament selection on diversity recovery and maintenance. In *GECCO '16 Companion: Proceedings of the Companion Publication of the 2016 Annual Conference on Genetic and Evolutionary Computation*, pages 983–990, Denver, Colorado, USA, 20-24 July 2016. ACM.

[35] Lee Spector, Nicholas Freitag McPhee, Thomas Helmuth, Maggie M. Casale, and Julian Oks. Evolution evolves with autoconstruction. In *GECCO '16 Companion: Proceedings of the Companion Publication of the 2016 Annual Conference on Genetic and Evolutionary Computation*, pages 1349–1356, Denver, Colorado, USA, 20-24 July 2016. ACM.

[36] Nicholas Freitag McPhee, Maggie M. Casale, Mitchell Finzel, Thomas Helmuth, and Lee Spector. Visualizing genetic programming ancestries. In *GECCO '16 Companion: Proceedings of the Companion Publication of the 2016 Annual Conference on Genetic and Evolutionary Computation*, pages 1419–1426, Denver, Colorado, USA, 20-24 July 2016. ACM.

[37] Pawel Liskowski, Krzysztof Krawiec, Thomas Helmuth, and Lee Spector. Comparison of semantic-aware selection methods in genetic programming. In *GECCO 2015 Semantic Methods in Genetic Programming (SMGP'15) Workshop*, pages 1301–1307, Madrid, Spain, 11-15 July 2015. ACM.

[38] Thomas Helmuth and Lee Spector. Evolving a digital multiplier with the PushGP genetic programming system. In *GECCO '13 Companion: Proceeding of the fifteenth annual conference companion on Genetic and evolutionary computation conference companion*, pages 1627–1634, Amsterdam, The Netherlands, 6-10 July 2013. ACM.

[39] Thomas Helmuth, Lee Spector, and Brian Martin. Size-based tournaments for node selection. In *GECCO 2011 Graduate students workshop*, pages 799–802, Dublin, Ireland, 12-16 July 2011. ACM.

Non-Refereed Workshop Publications

[40] Edward Pantridge and Thomas Helmuth. Code building genetic programming is fast. In Wolfgang Banzhaf, Ting Hu, Alexander Lalejini, and Stephan Winkler, editors, *Genetic Programming Theory and Practice XXI*, Genetic and Evolutionary Computation, University of Michigan, USA, June 6-8 2024.

[41] Edward Pantridge, Thomas Helmuth, and Lee Spector. Comparison of linear genome representations for software synthesis. In Wolfgang Banzhaf, Erik Goodman, Leigh Sheneman, Leonardo Trujillo, and Bill Worzel, editors, *Genetic Programming Theory and Practice XVII*, pages 255–274, East Lansing, MI, USA, 16-19 May 2019. Springer.

[42] Sarah Anne Troise and Thomas Helmuth. Lexicase selection with weighted shuffle. In Wolfgang Banzhaf, Randal S. Olson, William Tozier, and Rick Riolo, editors, *Genetic Programming Theory and Practice XV*, Genetic and Evolutionary Computation, pages 89–104, University of Michigan in Ann Arbor, USA, May 18–20 2017. Springer.

[43] Lee Spector, William La Cava, Saul Shanabrook, Thomas Helmuth, and Edward Pantridge. Relaxations of lexicase parent selection. In Wolfgang Banzhaf, Randal S. Olson, William Tozier, and Rick Riolo, editors, *Genetic Programming Theory and Practice XV*, Genetic and Evolutionary Computation, pages 105–120, University of Michigan in Ann Arbor, USA, May 18–20 2017. Springer.

[44] Thomas Helmuth, Lee Spector, Nicholas Freitag McPhee, and Saul Shanabrook. Linear genomes for structured programs. In Rick Riolo, Bill Worzel, Brian Goldman, and Bill Tozier, editors, *Genetic Programming Theory and Practice XIV*, pages 85–100, Ann Arbor, USA, 19-21 May 2016. Springer.

[45] Nicholas Freitag McPhee, Mitchell D. Finzel, Maggie M. Casale, Thomas Helmuth, and Lee Spector. A detailed analysis of a PushGP run. In Rick Riolo, Bill Worzel, Brian Goldman, and Bill Tozier, editors, *Genetic Programming Theory and Practice XIV*, pages 65–83, Ann Arbor, USA, 19-21 May 2016. Springer.

[46] Thomas Helmuth, Nicholas Freitag McPhee, and Lee Spector. Lexicase selection for program synthesis: A diversity analysis. In *Genetic Programming Theory and Practice XIII*, Genetic and Evolutionary Computation, Ann Arbor, USA, May 2015. Springer.

[47] Nicholas Freitag McPhee, David Donatucci, and Thomas Helmuth. Using graph databases to explore genetic programming run dynamics. In *Genetic Programming Theory and Practice XIII*, Genetic and Evolutionary Computation, Ann Arbor, USA, May 2015. Springer.

[48] Karthik Kannappan, Lee Spector, Moshe Sipper, Thomas Helmuth, William La Cava, Jake Wisdom, and Omri Bernstein. Analyzing a decade of human-competitive ("HUMIE") winners: What can we learn? In *Genetic Programming Theory and Practice XII*, Genetic and Evolutionary Computation, pages 149–166, Ann Arbor, USA, 8-10 May 2014. Springer.

[49] Lee Spector and Thomas Helmuth. Uniform linear transformation with repair and alternation in genetic programming. In *Genetic Programming Theory and Practice XI*, Genetic and Evolutionary Computation, chapter 8, pages 137–153. Springer, Ann Arbor, USA, 9-11 May 2013.

[50] Thomas Helmuth and Lee Spector. Evolving SQL queries from examples with developmental genetic programming. In *Genetic Programming Theory and Practice X*, Genetic and Evolutionary Computation, chapter 1, pages 1–14. Springer, Ann Arbor, USA, 12-14 May 2012.

[51] Lee Spector, Kyle Harrington, Brian Martin, and Thomas Helmuth. What's in an evolved name? the evolution of modularity via tag-based reference. In *Genetic Programming Theory and Practice IX*, Genetic and Evolutionary Computation, chapter 1, pages 1–16. Springer, Ann Arbor, USA, 12-14 May 2011.

Refereed Posters and Poster Papers

[52] Ryan Boldi, Ashley Bao, Martin Briesch, Thomas Helmuth, Dominik Sobania, Lee Spector, and Alexander Lalejini. A comprehensive analysis of Down-sampling for genetic Programming-based program synthesis. In *Proceedings of the 2024 Genetic and Evolutionary Computation Conference*

Companion, GECCO '24 Companion, pages 487–490, New York, NY, USA, 2024. Association for Computing Machinery.

[53] Ryan Boldi, Alexander Lalejini, Thomas Helmuth, and Lee Spector. A static analysis of informed Down-Samples. In *Proceedings of the 2023 Genetic and Evolutionary Computation Conference Companion*, GECCO '23, pages 531–534, Lisbon, Portugal, 15-19 July 2023. Association for Computing Machinery.

[54] Anil Kumar Saini, Lee Spector, and Thomas Helmuth. Environments with local scopes for modules in genetic programming. In *Proceedings of the 2022 Genetic and Evolutionary Computation Conference Companion*, GECCO '22, Boston, USA, July 9-13 2022. ACM.

[55] Li Ding, Ryan Boldi, Thomas Helmuth, and Lee Spector. Going faster and hence further with lexicase selection. In *Proceedings of the 2022 Genetic and Evolutionary Computation Conference Companion*, GECCO '22, Boston, USA, July 9-13 2022. ACM.

[56] Thomas Helmuth and Amr Abdelhady. Benchmarking parent selection for program synthesis by genetic programming. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion*, GECCO '20, pages 237–238. ACM, July 8-12 2020.

[57] Thomas Helmuth, Lee Spector, and Edward Pantridge. Counterexample-driven genetic programming without formal specifications. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion*, GECCO '20, pages 239–240. ACM, July 8-12 2020.

[58] Thomas Helmuth, Edward Pantridge, Grace Woolson, and Lee Spector. Transfer learning of genetic programming instruction sets. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion*, GECCO '20, pages 241–242. ACM, July 8-12 2020.

[59] Nicholas Freitag McPhee, Thomas Helmuth, and Lee Spector. Using algorithm configuration tools to optimize genetic programming parameters: A case study. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, GECCO '17, pages 243–244, Berlin, Germany, 15-19 July 2017. ACM.

[60] Nicholas Freitag McPhee, Maggie M. Casale, Mitchell Finzel, Thomas Helmuth, and Lee Spector. Visualizing genetic programming ancestries using graph databases. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, GECCO '17, pages 245–246, Berlin, Germany, 15-19 July 2017. ACM.

[61] Lee Spector and Thomas Helmuth. Effective simplification of evolved push programs using a simple, stochastic hill-climber. In *GECCO Comp '14: Proceedings of the 2014 conference companion on Genetic and evolutionary computation companion*, pages 147–148, Vancouver, BC, Canada, 12-16 July 2014. ACM.

[62] Thomas Helmuth and Lee Spector. Empirical investigation of size-based tournaments for node selection in genetic programming. In *GECCO Companion '12: Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference companion*, pages 1485–1486, Philadelphia, Pennsylvania, USA, 7-11 July 2012. ACM.

[63] Lee Spector, Thomas Helmuth, and Kyle Harrington. Fecundity and selectivity in evolutionary computation. In *Proceedings of the 13th Annual Conference Companion on Genetic and Evolutionary Computation*, GECCO '11, pages 129–130, Dublin, Ireland, 12-16 July 2011. ACM.

Tutorial with Published Slides

[64] William La Cava and Thomas Helmuth. Lexicase selection. In *Proceedings of the Conference on Genetic and Evolutionary Computation Companion*, GECCO '23, page 976–989, New York, NY, USA, 2023. Association for Computing Machinery.

[65] Thomas Helmuth and William La Cava. Lexicase selection. In *Proceedings of the 2021 Genetic and Evolutionary Computation Conference Companion*, GECCO '22. ACM, July 9-13 2022.

[66] Thomas Helmuth and William La Cava. Lexicase selection. In *Proceedings of the 2021 Genetic and Evolutionary Computation Conference Companion*, GECCO '21. ACM, July 10-14 2021.

Technical Reports

[67] Thomas Helmuth and Lee Spector. Detailed problem descriptions for general program synthesis benchmark suite. Technical Report UM-CS-2015-006, Computer Science, University of Massachusetts, Amherst, June 2015.

Invited Talks and Panels

Moderated AI at Hamilton Panel
Hamilton College *March and April 2024*
Moderated two panels on the use of artificial intelligence and machine learning by Hamilton faculty in research and teaching.

Evolving Software that Generalizes
Colgate University *March 2018*

Teaching Experience

Hamilton College

Computer Science for All **CS 101**
(F20, S21, S22)
I helped redesign our introductory course when we changed our curriculum in Fall 2020. This course introduces computer science for majors and non-majors. Course introduces programming in Python, including data types, basic data structures, control, and functions. These concepts are motivated by problem domains from across the liberal arts spectrum, from audio and image manipulation to natural language processing and web scraping.

Design Principles **CS 102**
(S24)
A second course in programming, concentrating on principled software design techniques. Topics include class design, effective functional decomposition, recursion, and structured data. Students will continue to develop programming proficiency by writing programs in the C++ programming language. Course discussion will emphasize efficient implementations in terms of memory space and running time, computational complexity of algorithms, and an introduction to data structures.

Introduction to Computer Science *(discontinued curriculum)* **CS 110**

(F17, S19, S20)

Introductory computer science course for majors and non-majors. Course introduces programming in Python, including basic data structures, control, functions, graphics, and graphical user interfaces.

Applied Theory *(discontinued curriculum)* **CS 210**

(F18, S20, F21)

An investigation of the nature of computation through development of several models of computation, from finite automata to Turing machines. This course covers some of the biggest ideas in computing, such as the existance of problems that are not solvable by any algorithm and how to show that there is likely no efficient algorithm to solve a problem.

Principles of Programming Languages **CS 220**

(S18, F18, S23, F24)

Students are exposed to a range of programming paradigms through learning four new programming languages. Students also complete a term-long research project, during which they learn a language on their own and present it in a paper and presentation.

Algorithms and Data Structures **CS 230**

(F22, S24, F24)

A study of fundamental algorithms and data structures from the canon, emphasizing correctness, and trade-offs between techniques, alongside deliberate coverage of topics from discrete mathematics. Topics include running time and space analysis, invariants as a means of proving correctness, and the divide-and-conquer, greedy, and dynamic programming algorithm design paradigms.

Genetic Programming **CS 307/411**

(F17, F20, F21, F23)

I developed this course. Genetic programming borrows concepts from biological evolution in order to evolve populations of computer programs. In this course, students will learn the genetic programming algorithm, implement it in Clojure, and conduct term projects to add functionality to their implementations. Students will read and review recent research articles in the field. Students are not expected to know Clojure prior to the course, and will gain a deeper understanding about the functional programming paradigm during the course.

Artificial Intelligence **CS 375**

(S19, S20, S21, S22, F22, S25)

Introduction to the theory and implementation of artificial intelligence. This course covers both foundational and modern approaches to AI, and explores a common thread of searching intelligently for solutions. Students will learn to select an appropriate AI representation to solve a problem and empirically analyze the performance of AI systems. Topics include heuristic search, game playing, evolutionary computation, machine learning, and the ethics of artificial intelligence.

Theory of Computation **CS 380**

(F23)

I developed this course. This course provides a thorough examination of the fundamental concepts underpinning all of computation. The course introduces the ideas of formal languages and computational models, and uses them to discuss computability and the existence of unsolvable problems. We develop increasingly complex models of computation, including finite automata, regular expressions, context-free grammars, and Turing machines. The course concludes with a high-level discussion of the computational complexity classes P and NP.

Washington and Lee University

Fundamentals of Programming I **CSCI 111**

(F15, F16)

Introductory computer science course for majors and non-majors. Course introduces programming in Python, including basic data structures, control, functions, graphics, and graphical user interfaces.

Algorithm Analysis **CSCI 211**

(W17)

A theory-based algorithms course covering algorithmic complexity analysis and a variety of common types of algorithms, including greedy, divide and conquer, and dynamic programming. Additionally covers basic graph algorithms, network flow, and advanced data structures.

Genetic Programming **CSCI 253**

(S16)

I developed this course. Genetic programming borrows concepts from biological evolution in order to evolve populations of computer programs. In the course, students learn the programming language Clojure and use it to implement a genetic programming system, adapting it to solve a problem of their choice.

Programming Language Design **CSCI 312**

(F15, F16)

I developed a new format for this course. Students are exposed to a range of programming paradigms through learning four new programming languages. Students also complete a term-long research project, during which they learn a language on their own and present it in a paper and presentation.

Theory of Computation **CSCI 313**

(W16, W17)

Introduces the fundamentals of theoretical computer science. Topics include abstract machines (finite automata, context-free grammars, and turing machines), decidability, and computational complexity.

Hampshire College

Principles of Programming Languages **Cognitive Science 286**

(S13), Instructor

I developed and taught this course. Students entered course with a wide range of backgrounds. This course introduced students to multiple programming paradigms through learning new programming languages.

Research Mentor

Honors Theses and Senior Projects, Hamilton College

James Frazier '24 *(senior fellowship)*
Applications of Artificial Intelligence to Information Privacy *2023-24*

Emmett Erickson '22
A Virtual Pilgrimage: Exploring Our Sense of Place and Feeling in Virtual Reality *2022*

Conor Courtney '20
Evolving Programs: Applying Biological Concepts to a Genetic Programming System *2020*

John (Jack) Hay '19
PoolGP: Evolving Pool Players Using Vector Operations *2019*

Lia Jundt '18
Combining novelty search and lexicase selection *2018*

Alex Dennis '18
Derivation of objectives for lexicase selection *2018*

Summer Research, Hamilton College

Jayden Fedoroff '25 and Kien Tran '25
Improving Code-Building Genetic Programming. *Summer 2023*

Ahmed Abdelrehim '24, James Frazier '24, and Violet Shi '24
Human-driven genetic programming. *Summer 2022*

Justin Marler '22 and Qianzi Hou '23
Adapting the genetic source in genetic programming. *Summer 2021*

Amr Abdelhady '21, Peter Kelly '21, and Grace Woolson '21
Understanding and benchmarking genetic programming. *Summer 2019*

Jack Scacco '21 and Xingyu He '21
New instructions and operators for genetic programming. *Summer 2018*

Summer Research, Washington and Lee University

Madubuko Chiedozie '20 and Rajwol Joshi '18
Program synthesis benchmark problems for genetic programming. *Summer 2017*

Sarah Anne Troise '19 and Hammad Ahmad '19
Weighted lexicase selection and lexicase-based initialization in genetic programming. *Summer 2016*

Departmental Service

Chair of Computer Science Department *Fall 2024–present*

Organized TA Hiring *Fall 2020–Fall 2024*

Chair of Computer Science Search Committee
Two positions *Fall 2024*

Visiting Computer Science Search Committee
Two visiting positions *Fall 2024–Spring 2025*

Chair of Visiting Computer Science Search Committee
Two visiting positions *Fall 2023–Spring 2024*

Computer Science Search Committee
Two positions *Fall 2023–Spring 2024*

Computer Science Search Committee
Two positions *Fall 2022–Spring 2023*

Computer Science Search Committee
Two positions *Fall 2021–Spring 2022*

Computer Science Search Committee
One tenure-track position and one visiting position *Fall 2020-Spring 2021*

Computer Science Search Committee
One tenure-track position and one visiting position *Fall 2018–Spring 2019*

Computer Science Search Committee
Two positions *Fall 2017–Spring 2018*

ACM ICPC Programming Contest
Helped organize contest and Hamilton's teams. Judged the contest in 2018. *Fall 2017, 2018, 2019*

Computer Science Colloquium Lunch Series
Organizer *2018-2019*

College Service

Alumni Council *2022–present*

AHA! Group: STEM Active Learning Lunch Principal Investigator *2018–2019, 2020–2021*

Committee for Library and IT Services
Digital Hamilton subcommittee in 2020 *2018-2021*

QuestBridge Scholar Advisor *2020–2021*

Sigma Xi Executive Committee *2020–2024*

Scottish Country Dancing Club Faculty Advisor *2017–present*

Professional Service

Editorial Board Member

Genetic Programming and Evolvable Machines *2021–present*
Associate Editor *2025–present*

Journal/Conference Reviewer

Genetic and Evolutionary Computation Conference (GECCO) *2018–2024*

AAAI *2021–2022*

Genetic Programming and Evolvable Machines *2012, 2014, 2015, 2018, 2019, 2021, 2022*

IEEE Trans. on Evolutionary Computation *2019–2022*

ACM Trans. on Evolutionary Learning and Optimization *2021–2022, 2024*

Neurocomputing *2016*

Data & Knowledge Engineering *2012*