

Christopher Kauffman

Associate Teaching Professor
Department of Computer Science
University of Minnesota 0571A
200 Union St SE Room 4-192
Minneapolis, MN 55455 USA

Office: 612-626-9351
Mobile: 612-396-4971
Email: kauffman@umn.edu
URL: <http://cs.umn.edu/~kauffman>

Education

Ph.D. Computer Science, University of Minnesota, Twin Cities. 2013.
Dissertation: Computational Methods for Protein Structure Prediction and Energy Minimization.

M.S. Computer Science, University of Minnesota, Twin Cities, 2010.

B.S. Computer Science, University of Minnesota, Twin Cities, 2004.
Minor in Mathematics. Graduated with High Distinction.

Teaching Experience

Teaching samples available at <http://cs.umn.edu/~kauffman/teaching-samples>

Associate Teaching Professor *Fall 2021 - Present*

Department of Computer Science, University of Minnesota, Twin Cities Minnesota.

- Full-time Teaching (80%) and Service (20%) faculty positions. Course load of 5 sections per year. Multi-year renewing contract-based appointment, granted via promotion process, approved by College of Science and Engineering Dean.
- Taught undergraduate and graduate courses on Machine Architecture (CSCI 2021, C and Assembly), Operating Systems (CSCI 4061, Unix/C), Parallel Computing (CSCI 5451) Advanced Functional Programming (CSCI 2041, OCaml), Discrete Math (CSCI 2011, Theory) and Introductory Programming (CSCI 1103, Java).
- Lectures sizes range from 30 to 250; focus on active learning strategies during lecture including in-class participation credit, live coding, and group problem sets. Out-of class work comprises comprehensive programming reflecting real-world applications.
- Provided department service on the Curriculum Committee, Computing/Web Infrastructure committee, contributed to Ethics Working group which planned incorporation of ethics topics into the CS major, liaison with the college IT department to ensure teaching technology needs of faculty are met.

Lecturer *Fall 2017 - Spring 2021*

Department of Computer Science, University of Minnesota, Twin Cities Minnesota.

- Full-time Teaching (100%) faculty positions. Course load of 5 sections per year. Contract-based appointment, optional participation in department service, no expectation of conducting research. Similar courses taught to current appointment.
- Participated in reforming the CSCI 2021 curriculum to adjust topic coverage according to feedback from students and the department undergraduate curriculum. Student perception of the course has improved markedly since these efforts.

- Enrolled and completed the 2018-2019 Early Career Teaching and Learning Program offered by the UMN Center for Educational Innovation. Discussed teaching strategies and improvement to pedagogy with a large group of faculty from various disciplines and a small group that was focused on STEM education.

Term Assistant Professor *Fall 2012 - Summer 2017*

Department of Computer Science, George Mason University, Fairfax, VA.

- Full-time, non-tenure track. Course load of 4 sections per semester, 2-3 preparations. Contract-based, no research responsibilities, service responsibilities related to undergraduate education.
- Taught courses on introductory programming (CS 211, Java), data structures (CS 310), computer ethics (CS 105), parallel programming (CS 499), programming for engineers (CS 222, C language), and CS for non-majors (CS 100). Contributed significantly to course development including new projects, labs, and grading rubrics.
- Class sizes range from 35 to 90; focus on active and engaging teaching strategies during lecture including in-class participation credit, live coding, and group problem sets.
- Developed two courses, CS 100: Principles of Computing (for non-majors) and CS 499: Parallel Computing (special topics for seniors).
- Led a CS teaching study group as part of an NSF funded program to improve engineering education.
- Mentored 6 undergrads on research projects through internal GMU funding sources.
- Active member of the Undergraduate Studies Committee to develop curriculum, primary author of CS honors program, honors college coordinator.
- Served as academic adviser for 65 students including 26 honors college students.

Instructor for CSC 301: Programming and Problem Solving *Fall 2011,*

Department of Mathematics, Concordia University, St. Paul.

Responsible for complete design and instruction of intro Java course for 15 students in math and science majors.

Instructor for CSCI 2011: Discrete Structures of Computer Science, *Summer 2011*

Department of Computer Science, University of Minnesota.

Responsible for instruction on mathematical concepts pertinent to computer science including lecture, assignments, and exams for 30 second-year students.

Summary of Student Ratings of Teaching from UMN

Course#	Instructor ¹	Course ²	#Rates ³	#Stdnts ⁴	#Sects ⁵	Course Title / Comment
CSCI 2021	5.69 / 6.00	5.48 / 6.00	1342	2255	17	Machine Architecture and Organization
CSCI 4061	5.72 / 6.00	5.62 / 6.00	175	629	5	Introduction to Operating Systems
CSCI 5451	5.22 / 6.00	4.96 / 6.00	72	107	2	Introduction to Parallel Computing
CSCI 2041	5.76 / 6.00	5.58 / 6.00	228	352	2	Advanced Programming Principles
CSCI 2011	5.45 / 6.00	5.58 / 6.00	28	37	1	Discrete Structures of Computer Science
CSCI 1103	5.66 / 6.00	5.58 / 6.00	62	144	1	Introduction to Java
All	5.67 / 6.00	5.48 / 6.00	2052	3876	27	Overall average of all ratings

- Results are averages over all students who provided ratings in a course.
- Instructor¹ summarizes questions related to the Instructor leading the course rated from 1 (worst) to 6 (best)
- Course² summarizes questions related to the Course rated from 1 (worst) to 6 (best)
- #Rates³ counts the total number of students who provided paper evaluations over all courses
- #Stdnts⁴ counts the total number of students who were enrolled each courses
- #Sects⁵ counts the total number of sections of a course which was rated

Summary of Teaching Ratings from GMU Student Evaluations

Course#	Teaching ¹	Course ²	#Rates ³	#Stdnts ⁴	#Sects ⁵	Course Title / Comment
CS 100	4.61 / 5.00	4.07 / 5.00	85	114	3	Principles of Computing
CS 105	4.77 / 5.00	4.24 / 5.00	568	654	19	Computer Ethics and Society
CS 211	4.78 / 5.00	4.53 / 5.00	140	299	6	Object-Oriented Programming
CS 222	4.75 / 5.00	4.46 / 5.00	105	145	4	Computer Programming for Engineers
CS 310	4.70 / 5.00	4.44 / 5.00	244	364	8	Data Structures
CS 499	4.84 / 5.00	4.84 / 5.00	25	33	1	Parallel Computing
All	4.74 / 5.00	4.34 / 5.00	1167	1609	41	Overall average of all ratings
Dept.	4.24 / 5.00	4.06 / 5.00				Overall Department Average ratings

- Results are averages over all students who provided ratings in a course.
- Teaching¹ measures answers to the prompt: "My overall rating of the teaching" rated from 1 (worst) to 5 (best)
- Course² measures answers to the prompt: "My overall rating of this course" rated from 1 (worst) to 5 (best)
- #Rates³ counts the total number of students who provided paper evaluations over all courses
- #Stdnts⁴ counts the total number of students who were enrolled each courses
- #Sects⁵ counts the total number of sections of a course which was rated

Service and Accomplishments

UMN CS Web and Computing Committee. Summer 2021-Present. Helped identify and resolve computing infrastructure deficits for the department. Acted as liaison to the College of Science and Engineering Web Team. Met with individual faculty to interview them about technologies they use and agglomerate results guide allocation of resources. Personally worked to update documentation and provide online tutorials to assist students with accessing resources.

UMN Linux Study Group Faculty Adviser. Summer 2022-Present. Along with Jack Kolb and local Kernel Developer Alex Elder, founded a student group focused on open source development and the internals of the Linux Kernel. Gave presentations on techniques associated with kernel development, set up computing infrastructure to enable students to experiment, organizing students to apply for official group status in Spring 2023.

UMN CS Ethics Integration Working Group. Fall 2021-Spring 2022. Helped determine how the department would integrated ethics topics into their curriculum. Brought prior experience in teaching computing ethics to the effort. Helped to design a variety of activities for our CS1 course to introduce ethics early and designed an ethics in engineering module for an Intro Operating Systems course.

UMN Mentoring Committee Chair for Jack Kolb. Fall 2021-Present. Provided guidance to new teaching professor on navigating the department, general curriculum, computing infrastructure. Worked closely with Dr. Kolb to ease the task of instructing several courses I have taught frequently in the past by discussing content, course assignment design, and best pedagogy.

UMN CS Curriculum Committee. Fall 2021-Spring 2022. Helped collect feedback on course sequencing for new Data Science major, on programming language changes in some courses, on course equivalences to other majors. Collaborated with Ethics Integration Working Group on forming new ethics requirements for CS majors. Helped with review of existing courses and offered recommendations for new ones.

Tool Time Lecture Series at the UMN, Spring 2020. Created and delivered a series of lectures discussing computing tools that don't fit well into most courses but are great practical utility. Lectures were delivered for a live audience, recorded, and posted for future students to study. <http://z.umn.edu/tooltime>

Curriculum Reform for UMN CSCI 2021, Spring 2018-Spring 2019. Met with Curriculum committee to discuss proposed course changes to CSCI 2021 to address student concerns and feedback. Implemented those changes over several offerings of the course in subsequent semesters and reported results back to curriculum committee.

UMN Early Career Teaching and Learning Program. Fall 2018 - Spring 2019. Participated in a variety of large- and small-group activities to improve classroom pedagogy. Small group focused on STEM-specific teaching issues.

Volunteer Evaluator for UMN EE1301 IoT Showcase. Spring/Fall 2019. Discussed student hardware/software projects at the completion of their intro EE programming course. Offered feedback and judging for projects involving Arduino compatible micro controllers.

Graduate School and Scholarship Recommendations at UMN. Fall/Spring 2018, 2019. Contributed recommendation letters for 30 undergraduate students and TAs as they apply graduate school, summer research experiences, and various scholarships, totaling 190 separate submissions to different schools.

GMU CS Undergraduate Studies Committee. Fall 2012 - Summer 2017. Active member since starting at GMU. Contributed to several specific goals including developing a CS departmental honors program (primary author), reforming curriculum to align with ACM 2013 and ABET recommendations, and giving recommendations to the university for General Education Information Technology learning outcomes.

GMU CS Web Committee. Fall 2013 - Summer 2017. Coordinated efforts to develop a new department homepage. Implemented several subsystems on the Django/Python site for syllabus upload, faculty directory, and technical reports display.

SIMPLE Teaching Group Leader for CS at GMU. Fall 2015 - Summer 2017. Small group meeting monthly to discuss education issues in in computer science including handling large classes, effective pedagogy, and improving diversity. Guided a discussion at a summit of group leaders and

members on being inclusive in courses while retaining rigor. SIMPLE was an NSF funded project at GMU to improve engineering education.

Faculty Adviser for GMU PatriotHackers Student Group. Fall 2013 - Summer 2017. Provided support for students interested in computer security including procuring equipment, advising on meeting topics, and recruiting members.

Journal Reviews for Bioinformatics (1 review), Proteins (1 review), Journal of Computational Chemistry (1 review), Theoretical Chemistry Accounts (1 review), Biomed Central Bioinformatics (3 reviews), IEEE Transactions on Knowledge and Data Engineering (2 reviews).

Refereed Conference, full paper reviews for European Conference on Computational Biology (ECCB), ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD), International Conference on Data Mining (ICDM), SIAM International Conference on Data Mining (SDM), European Conference on Machine Learning (ECML), International Parallel and Distributed Processing Symposium (IPDPS), Conference on Information and Knowledge Management (CIKM), International Symposium on Bioinformatics Research and Applications (ISBRA), International Conference on Genome Informatics (GIW).

DC STEM Fair Judge, 2014-2016. Judged a variety of K-12 student science projects for awards and advancement to higher fairs. Grand Awards Judge for advancement to the International Science and Engineering Fair twice.

Awards & Honors

Partnership for Affordable Content Grant Winner, Spring 2020, UMN Library. Awarded to develop instructional texts for computer science courses to be distributed to students freely thereby lowering costs for students and aligning study materials with the courses more closely.

Teaching Excellence Award Winner, Spring 2017, George Mason University. Nominated from 1,000+ faculty and selected from a pool of 30 applicants to be one of 8 winners of the annual, University-wide award. Special recognition for Technology-Enhanced Teaching.

<https://stearnscenter.gmu.edu/awards/teaching-excellence-award-winners/>

Computer Science Department Teaching Excellence Award, George Mason University, Spring 2015.

Nominee for George Mason University Teaching Excellence Award, Fall 2015.

Travel Grant to attend the European Conference on Computational Biology, September 2010, Ghent, Belgium (1200 Euros).

NIH Biotechnology Training Grant Trainee, June 2006 - June 2008 University of Minnesota. Selected graduate students received special training in new biotechnology developments including special lectures on sequencing technology, ethics, and interdisciplinary collaboration.

Japanese University Exchange, November 2007, Nara Institute of Science and Technology, Ikoma, Japan. After hosting several Japanese graduate students in computing and biological sciences, visited their university for one month to facilitate collaboration and idea exchange. <http://bsgcoe.naist.jp/en>

Software Projects

Testy: a program tester. <https://github.com/kauffman77/testy>

A testing “framework” designed to be useful for computing courses. It comprises a single file to download and run and is written in Shell Script so has no external dependencies save for standard Unix tools. I have used it in most of my courses successfully to provide fast student feedback as students work on programs. Being as simple as it is, it also integrates nicely into other contexts such as Gradescope’s Autograder.

Research Interests

Interplay of optimization theory and machine learning.

Protein structure prediction as an optimization/learning problem.

Applications of machine learning to bioinformatics.

Clustering programming projects to enable efficient grading.

Undergraduate Research Advisees and Projects at GMU

Abbreviations: OSCAR = Office of Student Scholarship, Creative Activities, and Research

URSP = Undergraduate Research Scholars Program

- 2016-2017 Ethan Rarity. OSCAR Federal Work Study. Support Vector Machine / R Interface Library (SVM/R). Focus on improving efficiency and flexibility of large dataset representation for machine learning in the R statistics language.
- 2016-2017 Hernan Ariascu. OSCAR Federal Work Study. DrJava IDE improvements to support CS course instruction in several GMU courses. Focus on improving quality of an excellent but orphaned IDE for beginning programming; improving compiler and unit test support. Resulted in a special edition with improvements which is used by GMU students: <https://cs.gmu.edu/~kauffman/drjava/>
- 2016 Xiaowen Fang. Independent project on clustering student programs to scale grading. Focus on using existing techniques for program plagiarism detection to cluster programs to ease the task of grading large numbers of student programs
- 2013-15 Cykielya Schultz, OSCAR Federal Work Study. AIDS Education Game Development. Focus on developing an interactive teaching game which demonstrates HIV infection at the molecular level, makes use of a Java-based game framework which can deploy desktop and web gaming, experience with Box2D physics engine. Collaboration with James Reid Schwebach in biology department.
- 2015-16 Vankhanh “Lilas” Dinh. OSCAR URSP Award. Lojban as an Intermediate for Natural Language Translation. Presented work at the National Conference on Undergraduate Research and the GMU/VSE Undergraduate Research Celebration.
- 2014 Saif Rizvi. OSCAR Federal Work Study. SVM / R Interface Library.
- 2013 Edward Martin. OSCAR URSP Award. AIDS Education Game Development.

Research Groups and Proposals

Proposal for an NSF REU Site on Educational Data Mining. 2015 and 2016. Co-author with Huzefa Rangwala of a proposal for a Research Experience for Undergraduates (REU) program to be held during summers at GMU. Students would focus on learning basic data mining and machine learning with applications to improving the quality of educational outcomes. Will instruct on machine learning tutorials and mentor several student projects.

SPARC Project. 2015 and 2016. Co-PI for Google funded project to increase retention and diversity in introductory programming courses while scaling up teaching capacity. Experimental sections focus on self-paced and peer-assisted learning during CS 1 and CS 2 courses. Project site: <https://sparc.cs.gmu.edu/>

Publications

Journal Papers and Book Chapters

- [1] **Chris Kauffman** and George Karypis. Ligand binding residue prediction. In *Introduction to Protein Structure Prediction: Methods and Algorithms*, Wiley Series in Bioinformatics. Wiley, 2010.
- [2] **Chris Kauffman** and George Karypis. LIBRUS: combined machine learning and homology information for sequence-based ligand-binding residue prediction. *Bioinformatics*, 25(23):3099–3107, 2009.
- [3] Huzefa Rangwala, **Christopher Kauffman**, and George Karypis. svmPRAT: SVM-based protein residue annotation toolkit. *BMC Bioinformatics*, 10:439, 2009.
- [4] **Christopher Kauffman** and George Karypis. Computational tools for protein-DNA interactions. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(1):14–28, 2012.
- [5] **Christopher Kauffman** and George Karypis. Coarse- and fine-grained models for proteins: evaluation by decoy discrimination. *Proteins*, 81(5):754–773, May 2013.

Conference Proceedings

- [1] Huzefa Rangwala, **Chris Kauffman**, and George Karypis. A kernel framework for protein residue annotation. In *Proceedings of the 13th Pacific-Asia Conference on Knowledge Discovery and Data-Mining*, 2009.
- [2] **Chris Kauffman**, Huzefa Rangwala, and George Karypis. Improving Homology Models for Protein-Ligand Binding Sites. In *LSS Computational Systems Bioinformatics Conference*, Stanford, CA, 2008.
- [3] **Chris Kauffman** and George Karypis. An Analysis of Information Content Present in Protein-DNA Interactions. In *Proceedings of the Pacific Symposium on Biocomputing*, pages 477–488, 2008.
- [4] Huzefa Rangwala, **Christopher Kauffman**, and George Karypis. A generalized framework for protein sequence annotation. In *Proceedings of the NIPS Workshop on Machine Learning in Computational Biology*, Vancouver, B.C., Canada., 2007.
- [5] Sigve Nakken, **Christopher Kauffman**, and George Karypis. Finding functionally related genes by local and global analysis of medline abstracts. In *SIGIR04 Bio Workshop: Search and Discovery in Bioinformatics*, Sheffield, UK, July 2004.

Conferences Attended

ACM Special Interest Group on Computer Science Education (SIGCSE), Minneapolis, Minnesota, 2019. Attendee only. Workshop on teaching track professor positions, CS for non-majors, sessions on effective assignments, ACM curriculum.

ACM Special Interest Group on Computer Science Education (SIGCSE), Memphis, Tennessee, 2016. Attendee only. Workshops on parallel computing and educational research.

ACM Special Interest Group on Computer Science Education (SIGCSE), Atlanta, Georgia, 2014. Attendee only. Focused on active teaching strategies and effective programming pedagogy.

Innovations in Teaching and Learning (ITL), George Mason University, Fairfax VA, Fall 2014, 2015, 2016. Presented a poster during 2016 on interactive teaching techniques used in programming classes.

European Conference on Computational Biology (ECCB), Ghent, Belgium, 2010. Poster presentation *A Convex Programming Model for Protein Structure Prediction*.

Computational Systems Bioinformatics Conference (CSB), Stanford, California, 2008. Paper presentation *Improving Homology Models for Protein-Ligand Binding Sites*.

Pacific Symposium on Biocomputing (PSB), Big Island, Hawaii, 2008. Poster presentation of paper: *An Analysis of Information Content Present in Protein-DNA Interactions*.

Critical Assessment of Techniques for Protein Structure Prediction (CASP), Pacific Grove, California, 2006. Student participant.

Invited Lectures

Protein Folding by Computer, by Hand, Spring 2016 guest lecture in CS 101: Preview of Computer Science. Fall 2013/2014 guest lecture in CS 390: Research and Project Design Principles in Computing. Discussed applications of machine learning and optimization to protein folding,

Freeing the Mind with the Machine, Fall 2016 invited talk for SRCT student group workshop on open source software development. Discussed reasons to get involved with open source software through use and code contributions.

Codons, Computers, and our Genomic Future, Fall 2011, invited talk for the Sigma Pi seminar series, Concordia University, St. Paul. Discussed how computation enables increased understanding of the genome.

Professional Experience

Cray Inc., Software Development Intern, Summer 2004 and 2005, Mendota Heights, MN. Worked in the parallel compiler development group on implementing a profile feedback mechanism and on process simulator enhancements.

Army High Performance Computing Research Center, Research Intern, Summer 2003. Developed software to handle global contact search for solid body simulations while gaining experience with parallel computing.

Minnesota Supercomputing Institute, Research Intern, Summer 2002, Minneapolis, MN. Extended features of a sparse matrix computations software package.