

LAMONT SAMUELS

University of Chicago
Department of Computer Science
5730 S Ellis Ave, Chicago, IL 60637
☎ (312)504-4942
✉ lamonts@cs.uchicago.edu

EDUCATION

- August 2016** **Ph.D., Computer Science, *University of Chicago***
TITLE Declarative Computer Graphics Using Functional Reactive Programming
RESEARCH Programming Languages, Computer Graphics, Functional Reactive Programming
- June 2013** **M.S., Computer Science, *University of Chicago***
TITLE Strand Communication in Diderot
RESEARCH Programming Languages, GPGPU, Image Analysis & Visualization
3.76 Systems GPA
- June 2010** **B.S., Computer Science, *California Polytechnic State University***
QUALIFICATIONS Bachelor of Science, *Cum Laude*
3.8 Major GPA, 3.64 overall

ACADEMIC EMPLOYMENT

- 07/18 - Present** **University of Chicago *Chicago, IL***
POSITION Assistant Clinical Professor
DUTIES Teach core system courses in the Masters Program of Computer Science (MPCS) and Computational Analysis & Public Policy (CAPP). Please see the “Teaching Experience” section for the courses I taught in this position. Additionally, I’m responsible for overseeing the core programming classes in MPCS and helping with the programming placement exam in the MPCS department.
- 07/19 - Present** **University of Chicago *Chicago, IL***
POSITION Associate Director, Masters Program in Computer Science
DUTIES Oversee the advising of students plan in regards to helping them plan their courses each quarter. Hold course planning sessions to provide an overview of the course for the upcoming quarter.
- 06/13 - 06/18** **University of Chicago *Chicago, IL***
POSITION Lecturer
DUTIES Taught an introductory Java course, and a functional programming course in the Masters of Computer Science program. I also taught an undergraduate C Programming course during the summer sessions. Please see the “Teaching Experience” section for the courses I taught in this position.
- 01/15 - 05/15** **School of the Art Institute of Chicago *Chicago, IL***
POSITION Lecturer
DUTIES Taught an introduction to computer science for art students. Designed and planned lessons focused towards providing the students with the fundamentals of programming and then allowing them to apply these skills with exercises and projects gear towards the computer graphics area.

TEACHING EXPERIENCE

COURSE **MPCS 52060: Parallel Programming** *University of Chicago*
TAUGHT Autumn 2018, Spring 2018, Autumn 2019, Spring 2020, Autumn 2020, and Spring 2021 (Currently)
POSITION Instructor
DUTIES The goal of this course is to provide an introduction to the foundations of parallel programming and to consider the performance gains and trade-offs involved in implementing and designing parallel computing systems. Specifically, this course will place an emphasis on concepts related to parallel programming on multicore processors. Programming exercises and projects come from the domains of AI and machine learning, computer graphics, cryptocurrency technologies, and scientific visualization. All assignments are implemented using the Go programming language.

COURSE **MPCS 51042: Python Programming** *University of Chicago*
TAUGHT Autumn 2018, Autumn 2019, Spring 2020, Autumn 2020, and Autumn 2020
POSITION Instructor
DUTIES This course provides a thorough overview of the Python 3 language with an emphasis on writing idiomatic code in Python and object-oriented design patterns and is suitable for students with some prior programming experience. Students develop an understanding of the core features of the languages and gain exposure to commonly used standard-library and third-party modules. Students received exposure to areas where Python is heavily used such as web development, scientific computing, graphical user interfaces, data mining, and systems programming.

COURSE **CAPP 30122: Computer Science with Applications II** *University of Chicago*
TAUGHT Winter 2019, Winter 2020, and Winter 2021
POSITION Instructor
DUTIES This course is the second in a three-quarter sequence that teaches computational thinking and skills to students from a wide-variety of fields. Lectures cover topics in (1) data representation, (2) relational databases, (3) data cleaning and presentation, (4) shell scripting, (5) data structures, such as graphs, hash tables, and heaps. Applications and datasets from a wide variety of fields serve both as examples in lectures and as the basis for programming assignments. In recent offerings, students have written a course search engine and a system to do speaker identification.

COURSE **MPCS 51400: Functional Programming** *University of Chicago*
TAUGHT Winter 2017, Autumn 2017, and Spring 2018
POSITION Instructor
DUTIES This course presents the functional programming paradigm. The course used the Haskell programming language. After learning the basic elements of these languages, students explored functional programming techniques that can be exploited in various areas of computer science.

COURSE **MPCS 51036: Java Programming** *University of Chicago*
TAUGHT Autumn 2016 and Spring 2017
POSITION Instructor
DUTIES This is a fast-paced first course in Java for students with some prior programming experience, though not necessarily Java or any other object-oriented language. A strong emphasis will be placed on understanding basic fundamentals of OO design—inheritance, polymorphism, composition, etc. and more generally on applying sound principles of contemporary software engineering and tools to real-world problems.

COURSE **CMSC 15200: Introduction to Computer Science II** *University of Chicago*
TAUGHT Summer 2013, and Summer 2014
POSITION Instructor
DUTIES The course provided an introduction to computer programming using the C language with an emphasis on developing general programming skills and familiarity with advance data structures. The most students in the course were non-computer science majors with a diverse mixture of discipline backgrounds (e.g., humanities, economics, mathematics). Average class sizes were 20 students for each course.

INDUSTRY EXPERIENCE

- 03/18 - 08/18** **BRD** *Chicago, IL*
POSITION Senior Software Engineer
DUTIES Implemented the Light Ethereum Subprotocol for the Ethereum blockchain in C. Designing and implementing client interfaces for iOS and Android that interact with their core Ethereum technology.
- 10/16 - 01/18** **Tanvas Inc.** *Chicago, IL*
POSITION Technical Lead (Software Team)
DUTIES Lead a team of four software engineers. Implemented a new generation of our C++ API and low-level driver for our hardware (~3k lines of code). Designed the new API and driver to be more modular to allow third-party developers to quickly prototype and integrate their systems/applications with our technologies and hardware. Architected the API and driver solution to also be modular and scalable to allow my team to easily add on new modules to be more agile and quickly prototype new features for our customers. Implemented the low-level driver to function on various platforms (Windows, Linux, Android).
- 11/15 - 16** **LensIQ** *Chicago, IL*
POSITION Lead Developer
DUTIES LensIQ was an innovative mobile application for identifying and providing useful information for progressive lens engravings. Presented LensIQ to a Fortune 500 company with three Senior Vice Presidents present for potential investment. Implemented the lens identification algorithm using the k-NN machine learning algorithm in OpenCV. Designed a custom-made user interface for the entire application in Adobe Photoshop. Incorporated the user-interface layout into separate Swift storyboards using Auto Layout and custom Swift classes. Developed the majority of the application, which entails coding 39 model-view-controller (MVC) classes (~3.5k lines of code) in Swift.
- 06/12 - 10/12** **AMD** *Sunnyvale, CA*
POSITION Research Co-Op
DUTIES Refactored Diderot's GPU target to run on their next generation graphics architecture to compare the performance to the previous generation. Debugged and developed benchmarks to test their next generation hardware.
- 06/09 - 06/10** **Collaborative Agent Design Research Center** *San Luis Obispo, CA*
POSITION Student Developer
DUTIES Reverse-engineered their previous cargo-planning software, a web application written in ActionScript, to record requirements that needed to be maintained within their next version. Developed behavioral tests based on predefined use cases from our clients. Implemented graphical user interfaces for their various cargo planning domains.
- 09/07 - 06/10** **Academic Skills Center** *San Luis Obispo, CA*
POSITION Study Session Group Leader
DUTIES Over seven quarters, led 17 groups of 150+ students in introductory programming courses within the College of Engineering. Answered questions and clarified concepts for students. Promoted student discussions and group bonding. Participated in staff training opportunities.
- 06/07- 09/07** **Clark County Government** *Las Vegas, NV*
POSITION GIS Programmer Intern
DUTIES Investigated and redesigned their web application, OpenWeb, which provides property information about county parcels using Javascript and AJAX. Implemented web service applications to allow customers to retrieve county maps and data sets. Examined various ways to incorporate ESRI's Web ADF into OpenWeb.

RESEARCH EXPERIENCE & PUBLICATIONS

As of 2017, I am not actively involved in research. However, my research interests have mostly revolved around programming languages, compilers, concurrent programming and GPGPU research. My current focus is primarily in teaching but below you will find my previous research work and publications as a graduate student.

11/14 - Current Tesel

TECH/LANGUAGE(S) Swift & Metal API & Standard ML of NJ

(TYPE) POSITION (Dissertation-Research) Main Developer

DESCRIPTION Tesel is an embedded domain-specific language in Swift that is used to develop efficient and declarative computer graphics applications using the Functional Reactive Programming paradigm.

DUTIES

- Devised and developed the language's constructs and combinators as a macOS framework to allow for easy integration into Xcode for programmers
- Constructed a runtime system written in Swift that includes a rendering system that uses the Metal API (~4k lines of code)
- Implemented a compiler written in Standard ML (~4k lines of code) that builds a standalone application using xcode-build and the Tesel runtime and framework.
- Provided parallelism mechanisms using Grand Central Dispatch (GCD) to allow programmers to execute portions of their code in parallel, which led to a 2x speedup in applications.
- Formalized the evaluation model of Tesel by providing a formal operational semantics of the language.

06/10 - 08/16 Diderot

TECH/LANGUAGE(S) Standard ML of NJ & C & OpenCL

(TYPE) POSITION (Team-based) Research Assistant

DESCRIPTION Diderot is a parallel domain-specific language that is designed for biomedical image-analysis and visualization algorithms and provides a high-level mathematical programming model.

DUTIES

- Programmed the initial GPU-backend in OpenCL using a persistent thread scheduler
- Formulated and built a spatial feature using a KDTree data structure that allows the local agents in language to communicate between each other.
- Implemented and designed global reductions operators (*e.g.*, sum, max, mean) for our parallel and sequential backends
- Integrated the features described above into our various stages compiler written in Standard ML

FULL PEER-REVIEWED PAPERS "Diderot: a Domain-Specific Language for Portable Parallel Scientific Visualization and Image Analysis." *VIS 2015*, C. Chiw, G. Kindlmann, J. Reppy, L. Samuels and N. Seltzer, January 2016.

"Diderot: A Parallel DSL for Image Analysis and Visualization," *PLDI 2012*, C. Chiw, G. Kindlmann, J. Reppy, L. Samuels and N. Seltzer, 2012.

POSTERS, WORKSHOPS OR SHORT PAPERS "Bulk-Synchronous Communication Mechanisms in Diderot." *AGERE!@SPLASH 2015*, J. Reppy and L. Samuels 2014.

"Bulk-Synchronous Communication Mechanisms in Diderot." *CPC 2014*, J. Reppy and L. Samuels 2014.

"Diderot: A Parallel DSL for Computing on Multi-Dimensional Tensor Fields." *VisWeek 2011*, G. Kindlmann, J. Reppy, L. Samuels and N. Seltzer, Poster session 2011.

TECHNICAL SKILLS

Languages: Proficient in: Python, Golang, C, Haskell, Swift

Experience with: Java, C++, Standard ML, HTML, Javascript, SQL

Technologies: OpenCL, CUDA, Metal API, Cocoa, Relational databases, FRP related libraries