

# Lai, Albert Yu Cheong

## Curriculum Vitae

(address and phone omitted in public)

trebla [at] vex [dot] net

Born in 1969

Citizenship: Canadian

## Interests

Program derivation methods (and tool support) such as refinement calculi, functional programming, programming contests, heuristics of mathematical arguments (and tool support).

Pedagogic mottos: Teach by mentoring (fostering independence) rather than explaining (garnering subscriptions). Learn by investigative experimenting rather than complacent consuming. Concrete before abstract, specific before general (from Piaget's theory).

## Education

PhD Computer Science, University of Toronto, 2013. Supervised by E. C. R. Hehner.

NASA Summer School in Engineering Theories of Software Construction,  
Marktoberdorf, Germany, 2000.

MSc Computer Science, University of Toronto, 2000. Supervised by E. C. R. Hehner.

BSc(H) Mathematics, University of Toronto, 1996.

BSc(H) Information Technology, City University of Hong Kong, 1992.

## Teaching

Instructor of Principles of Programming Languages

CSCC24 (3rd year), University of Toronto Scarborough

2025-2018 (11 times)

CSC324 (3rd year), University of Toronto St. George

2020 Fall 2nd-half; 2019 Winter

- Introduction to functional programming via Haskell. Syntax: Context-free grammars and parsing. Semantics: Evaluation strategies (call by value, lazy); interpreters for (executable math models of) local bindings (thus environments), functions (thus closures), call by value, mutable variables, continuations. Type inference.

Teaching assistant of Principles of Programming Languages

CSCC24 (3rd year), University of Toronto Scarborough

2023 Winter; 2022 Winter

- Marking. Conducting tutorials (2023). Contributing to exam questions (2022).
- Instructor Anya Tafliovich.

Instructor of Software Tools And Systems Programming

CSCB09 (2nd year), University of Toronto Scarborough

2025-2019 (7 Summers)

- Unix shell scripting, utilities, C programming, system programming—e.g., processes, pipes, sockets, signals.

Instructor of Design and Analysis of Data Structures

CSCB63 (2nd year), University of Toronto Scarborough

Winters 2025-2024; Summers 2018-2014; Winter 2018

Instructor of Data Structures and Analysis

CSC263 (2nd year), University of Toronto St. George

Summer 2015

- Common data structures and their algorithms (balanced binary search trees, heaps, union-find, hash tables, B-trees). Some basic graph algorithms (depth-first search, breadth-first search, minimum spanning tree, single-source shortest path). Asymptotic worst-case time, amortized time, and average-case time.

Teaching assistant of Computer Graphics

CSCD18 (4th year), University of Toronto Scarborough

Fall 2018

- Instructor Francisco Estrada.

Teaching assistant of Computability and Computational Complexity

CSCC63 (3rd year), University of Toronto Scarborough

Winters 2016-2014 (3 times)

- Turing machines, computability, NP-completeness, PSPACE, (N)LOGSPACE
- Conventional tutorials and marking
- Instructor Michael Molloy

Teaching assistant of Introduction to the Theory of Computation

CSCB36 (2nd year), University of Toronto Scarborough

Fall 2012, Summer 2012

- Induction proofs, program correctness proofs (e.g., loop invariants), formal logic (syntax and semantics), regular languages, context-free languages.
- Tutorials are inverted classroom: students do exercises in small groups, I go around offering personal help; sometimes some student solutions are presented to the whole class. (Fall 2012)
- Part of student grades comes from a dedicated web forum, where students earn marks by both creating questions (scope dictated by instructor) and answering questions. The instructor and I are moderators and graders.
- Instructor Nick Cheng

Teaching assistant of Computability and Computational Complexity

CSCC63 (2nd year), University of Toronto Scarborough

Summer 2012

- Turing machines, computability, NP-completeness
- (Tutorials are inverted classroom like above, but I wasn't assigned to it yet.)
- Part of student grades comes from a dedicated web forum, where students earn marks by both creating questions (scope dictated by instructor) and answering questions. The instructor and I are moderators and graders.

- Instructor Nick Cheng

Instructor (Course Director) of Introduction to Program Verification  
CSE3341 (3rd year), York University  
Winters 2009, 2006-2003

- Program correctness and derivation by Hehner's predicative refinement

Instructor of Formal Methods in Software Design  
CSC465 (4th year), University of Toronto  
Fall 2008

- Program correctness and derivation by Hehner's predicative refinement

Teaching assistant of Formal Methods in Software Design  
CSC465 (4th year and graduate), University of Toronto  
Falls 2007-2003, 2001-1999

- Program correctness and derivation by Hehner's predicative refinement
- instructor E.C.R. Hehner

Head teaching assistant of Introduction to Computer Science  
CSC148 (1st year), University of Toronto  
Summers 2004-2002

- Duties: automarking programs (and resulting Q&A with students), grade collection, printing and distribution.
- Instructor Daniel Wigdor

Teaching assistant of Software Engineering II  
ECE450 (4th year), University of Toronto  
Winters 2001-2000

- Use of formal methods (computation tree logic, Hehner's predicative refinement) in software engineering.
- Instructors Marsha Chechik and E.C.R. Hehner

Teaching assistant of Introduction to Computer Science  
CSC148 (1st year), University of Toronto  
Summer 2000

- Instructor François Pitt

Teaching assistant of Software Tools in Unix and C  
CSCB09 (2nd year), University of Toronto Scarborough  
Winter 1997

- Using, scripting, and programming Unix systems.
- Instructor John Harper

Teaching assistant of Computer Basics  
CSCA06 (1st year), University of Toronto Scarborough  
Fall 1996

- Instructor Graeme Hirst

## **Publications**

### Theses

PhD Thesis: Eager, Lazy, and Other Executions for Predicative Programming. University of Toronto, 2013. (Finished in December 2012.) <http://www.cs.utoronto.ca/~trebla/albert-thesis-ut.pdf>

MSc Thesis: A Tool for A Formal Refinement Method. University of Toronto, 2000. (Finished in January 2000.) <http://www.cs.toronto.edu/~trebla/scphEditor/>

### Conferences

S. M. Easterbrook, M. Chechik, B. Devereux, A. Gurfinkel, A. Lai, V. Petrovykh, A. Tafliovich, and C. Thompson-Walsh

XChek: A Model Checker for Multi-Valued Reasoning

in Proceedings, 25th International Conference on Software Engineering, Portland, Oregon, May, 2003.

M. Chechik, B. Devereux, S. Easterbrook, A. Lai, and V. Petrovykh

Efficient Multiple-Valued Model-Checking Using Lattice Representations

in Proceedings of 12th International Conference on Concurrency Theory (CONCUR'01), Aalborg, Denmark, 451-465, August, 2001.

### Technical reports

M. Chechik, A. Gurfinkel, B. Devereux, A. Lai, and S. Easterbrook

Data Structures for Symbolic Multi-Valued Model-Checking

CSRG Technical Report, Department of Computer Science, University of Toronto, January, 2002.

M. Chechik, S. Easterbrook, B. Devereux, A. Lai, V. Petrovykh and C. Thompson-Walsh

A Framework for Multi-Valued Reasoning over Inconsistent Viewpoints—Project Report

CSRG Technical Report, Department of Computer Science, University of Toronto, August, 2000.

### Teaching

Numerous Haskell articles at <http://www.vex.net/~trebla/haskell/>

Some mathematics articles at <http://www.vex.net/~trebla/homework/>

## **Honors and Awards**

NSERC Postgraduate Scholarship, 1996-1998; 1999-2001

U of Toronto Scarborough Graduation Prize in Physical Sciences, 1996

U of Toronto Scarborough A. D. Allen Scholarship, 1995

University of Toronto Scholar, 1995

NSERC Undergraduate Student Research Award, 1995

U of Toronto Samuel Beatty Award, 1995

University of Toronto Scholar, 1994

NSERC Undergraduate Student Research Award, 1994

U of Toronto Samuel Beatty Award, 1994

## Service

Internet Relay Chat channel operator (moderator) of Freenode #haskell (2009 - present) and EFNet #compsci, #math, and #physics (circa 2000 - present); maintainer of #compsci homepage <http://www.vex.net/~trebla/compsci/index.html> (circa 2005 - present).

Referee:

Formal Aspects of Computing, September 2009.  
Journal of Logic and Algebraic Programming, March 2009.  
UTP, June 2008.  
Acta Informatica, June 2006.  
(sub-referee) CONCUR, May 2005.  
IEEE Transactions on Software Engineering, August 2005.  
Journal of Systems and Software, December 2000.

Coaching of teams for the ACM Inter-Collegiate Programming Contest

University of Toronto Scarborough: present-10/2019

University of Toronto St. George: 03/2006-10/2003, 03/2001-10/2000, (coach emeritus) 03/2002-10/2001, (assistant coach) 2001-1997.

Judge of the local programming contest: the University of Toronto Inter-Campus Programming Contest, 2002-1997.

Co-Chair of the University of Toronto ACM Student Chapter, 2000-1998;  
Secretary of same, 2001.

Organizer of the annual area meeting for Programming Languages and Systems, Department of Computer Science, University of Toronto, 2003-1997.

- Faculty in the area introduce themselves and their research interests to new graduate students

## Computing Skills

Small programming portfolio at <http://www.vex.net/~trebla/portfolio/> and <https://github.com/treblacy>

Fluent in Haskell, C, shell scripting with bash.

Comfortable with Unix-like environments (including user-end operation, system programming, and administration), Racket Scheme, Lean, Prolog, Javascript, C++, Java, LaTeX.

Exposed to Eclipse plugin writing, Lisp, Scala, OCaml, SML, Python, SQL, XSLT 1.0, DTD for XML, Windows user-end operation, OpenOffice and LibreOffice, TeX.