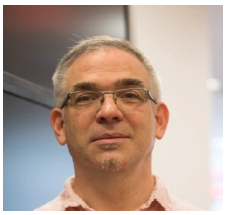

Dr. Gregory V. Wilson

65 Highfield Road
Toronto, Ontario, Canada
gvwilson@third-bit.com
http://third-bit.com
416 435 9779

Employment

2021–present	Senior Engineering Manager, Deep Genomics. Responsibilities include recruiting, managing junior developers, design and implementation of back-end application software in Python, and training.
2021	Head of Education, Metabase. Responsible for designing and delivering training material and managing content development team.
2018–2021	Data Scientist and Professional Educator, RStudio PBC. Created and ran an instructor certification program that trained almost 200 people in its first two years; also responsible for the summer intern and student mentorship programs.
2017–2018	Content developer and instructor trainer, DataCamp. Created courses on Git and the Unix shell; recruited, trained, and edited the work of freelance instructors.
2017	Principal Consultant, Rangle.io. Revised training materials on Angular and React; coached company staff on training techniques.
2015–2016	Director of Instructor Training, Software Carpentry Foundation. Developed and delivered the foundation's train-the-trainers course; helped develop workflow tools used to manage thousands of volunteer instructors worldwide.
2012–2015	Executive Director, Software Carpentry Foundation. Developed curriculum, trained instructors, negotiated partnerships with multiple organizations, and led development of workflow tools.
2011	Software Engineer, Side Effects Software Inc. Helped build and test a web store for the company's flagship product using Django and Selenium.
2010–2011	Project lead, Software Carpentry. Developed and delivered workshops on research computing skills at several dozen universities; recruited and trained volunteer instructors; oversaw program assessment and fundraising.
2006–2010	Assistant Professor, Dept. of Computer Science, University of Toronto. Taught graduate/undergraduate courses; supervised theses; developed a Professional Master's degree program.
2004–2006	Independent consultant. Wrote a book on data crunching in Python; rewrote the Software Carpentry course under a grant from the Python Software Foundation; developed and taught courses in Computer Science at the University of Toronto.
2000–2004	Senior software engineer, Baltimore Technologies (acquired by Hewlett Packard). Helped develop a single sign-on product using C++ and Java on Linux and Windows. Also taught courses and supervised undergraduate honors projects at the University of Toronto.

1998–2000	Independent consultant. Ran Software Carpentry classes at Los Alamos National Laboratory; helped develop a single sign-on product for Nevex Software (acquired by Baltimore Technologies); co-designed Python's set module.
1982–1998	Worked as a software developer for firms ranging from early-stage startups to IBM, including six years as a research software engineer at the Edinburgh Parallel Computing Centre; wrote and edited books on parallel programming.

Education

1993	PhD in Computer Science, University of Edinburgh. Thesis was *Structuring and Supporting Programs on Parallel Computers*.
1986	MSc in Artificial Intelligence, University of Edinburgh. Thesis was *An Implementation of a Connection Method Theorem Prover for S5 Modal Logic*.
1984	BSc in Mathematics and Engineering (First Class Honors), Faculty of Applied Science, Queen's University, Ontario.

Awards

- ACM SIGSOFT Influential Educator of the Year Award, 2020.
- ComputerWorld Canada's "IT Educator of the Year" award, 2010.
- Co-winner of 2008 Jolt Award for Best General Book (for *Beautiful Code*).
- University of Toronto Computer Science Student Union Teaching Award, 2004.
- Co-winner of Howe Prize (best thesis in Artificial Intelligence), University of Edinburgh, 1986.
- Commonwealth Scholarship, 1985–86.
- University Medal, Queen's University, 1984 (top student in graduating class).
- Co-winner of A.B. Lillie Prize, 1984 (top student in Mathematics).

Other Achievements

- Author of two children's books (*Bottle of Light*, Scholastic Press Canada, 2008 and *Three Sensible Adventures*, Annick Press, 1999).
- Co-creator of AMY (a Django tool for coordinating workshops), TidyBlocks (a block-based environment for data science), and Glosario (a multilingual glossary of data science terms).
- Co-organized a summit meeting of free-range computing education groups in 2015.
- Co-founder and editor of It Will Never Work in Theory.
- Founder and co-editor of *The Architecture of Open Source Applications*.
- Python Software Foundation, 2010-present.
- Stencila Advisory Board, 2017-19.
- Toronto Public Library Innovation Council, 2017-18.
- Advisory Board, Ladies Learning Code, 2012-2014.
- Contributing editor with *Doctor Dobb's Journal*, 2001-10.
- Mentor for Google's Summer of Code, 2005-2015.
- Ultimate frisbee, 1995-2003 (Toronto "C" Division championship team 2002).
- Competed in World Computer Chess Championship, 1989.
- Past member/volunteer with the Canadian National Institute for the Blind, the Sierra Club, Amnesty International, OXFAM, the Bruce Trail Association, and the Green Party of Canada.

Technical Books

- Greg Wilson: *Software Design by Example*. Chapman and Hall/CRC Press, 2022.
- Damien Irving, Kate Hertweck, Luke Johnston, Joel Ostblom, Charlotte Wickham, and Greg Wilson: *Research Software Engineering with Python*. Chapman and Hall/CRC Press, 2021.
- Maya Gans, Toby Hodges, and Greg Wilson: *JavaScript for Data Science*. Chapman and Hall/CRC Press, 2020.
- Greg Wilson: *Teaching Tech Together*. Chapman and Hall/CRC Press, 2019.
- Amy Brown and Greg Wilson (eds.): *The Architecture of Open Source Applications* (two volumes), Lulu.com, 2011 and 2012.
- Andy Oram and Greg Wilson (eds.): *Making Software: What Really Works, and Why We Believe It*. O'Reilly, 2010.
- Jennifer Campbell, Paul Gries, Jason Montojo, and Greg Wilson: *Practical Programming*. Pragmatic Bookshelf, 2009.
- Andy Oram and Greg Wilson (eds.): *Beautiful Code: Leading Programmers Explain How They Think*. O'Reilly and Associates, 2007; winner of 2008 Jolt Award for Best General Book.
- Greg Wilson: *Data Crunching: Solve Everyday Problems Using Java, Python, and More.* Pragmatic Bookshelf, 2005.
- Gregory V. Wilson and Paul Lu (eds.): *Parallel Programming Using C++*. MIT Press, 1996.
- Gregory V. Wilson: *Practical Parallel Programming*. MIT Press, 1995.

Selected Papers and Articles

- Greg Wilson: "Twelve Quick Tips for Software Design". *PLoS Comp. Bio.*, 2022.
- Danielle Smalls and Greg Wilson: "Ten Quick Tips for Staying Safe Online". *PLoS Comp. Bio.*, 2021.
- Sarah Lin, Ibraheem Ali, and Greg Wilson: "Ten Quick Tips for Making Things Findable". *PLoS Comp. Bio.*, 2020.
- Paul Denny, Brett A. Becker, Michelle Craig, Greg Wilson, and Piotr Banaszkiewicz: "Research This! Questions that Computing Educators Most Want Computing Education Researchers to Answer". *ICER 2019*.
- Dan Sholler, Igor Steinmacher, Denae Ford, Mara Averick, Mike Hoye, and Greg Wilson: "Ten Simple Rules for Helping Newcomers Become Contributors to Open Projects". *PLoS Comp. Bio.*, 2019.
- Greg Wilson: "Ten Quick Tips for Creating an Effective Lesson". *PLoS Comp. Bio.*, 2019.
- Neil Brown and Greg Wilson: "Ten Quick Tips for Teaching Programming". *PLoS Comp. Bio.*, 2018.
- Gabriel Devenyi, Rémi Emonet, Rayna Harris, Kate Hertweck, Damien Irving, Ian Milligan, and Greg Wilson: "Ten Simple Rules for Collaborative Lesson Development". *PLoS Comp. Bio.*, 2018.
- Daniel Almeida, Gail Murphy, Greg Wilson, and Mike Hoye: "Do Software Developers Understand Open Source Licenses?" *ICSE'17*, 2017.
- Morgan Taschuk and Greg Wilson: "Ten Simple Rules for Making Research Software More Robust". *PLoS Comp. Bio.*, 2017.
- Greg Wilson: "Software Carpentry: Lessons Learned". *F1000 Research*, 2016.
- Greg Wilson, Dhavide Aruliah, Titus Brown, Neil Chue Hong, Matt Davis, Richard Guy, Steven Haddock, Kathryn Huff, Ian Mitchell, Mark Plumbley, Ben Waugh, Ethan White, and Paul Wilson: "Best Practices for Scientific Computing". *PLoS Biology*, 2014.
- Jo Erskine Hannay, Hans Petter Langtangen, Carolyn MacLeod, Dietmar Pfahl, Janice Singer, and Greg Wilson: "How Do Scientists Develop and Use Scientific Software?" *SECSE'09*, 2009.
- Jorge Aranda, Steve Easterbrook, and Greg Wilson: "Requirements in the Wild: How Small Companies Do It". *RE'07*, 2017.