landon rabern, phd

314 Euclid Ave.
Mt. Gretna, PA

(805) 403-8185
landon.rabern@gmail.com
https://github.com/landon

Work History

CTO, Co-founder   2008 - 2017
LBD Data

- Contract development for Safety Vision, a producer of mobile video solutions for police and public transit.
- Our software suite is used at police departments and transit authorities all over Texas, in Las Vegas, Honolulu, Indianapolis, Columbus, and Portland.
- Wrote SafetyView (C#)
  - P/Invoke wrapper of libavcodec for H264/MPEG4 decoding, up to 12 channel playback at once.
  - Vehicle location display via embedded Google map (JavaScript).
  - Custom AVI writer for export.
  - Shareable web links, multichannel playback in HTML5 for mobile (using Amazon S3 with DynamoDB).
  - Both WinForms and WPF versions.
- Wrote SafetyNet (C#)
  - Automatic wireless download of video from vehicles.
  - Video searchable by metadata in database (SQL).
  - Streaming video to SafetyView.
- Wrote LiveLook (C++ / C#)
  - Live playback of multiple IP camera streams.
  - Uses LIVE555 to eat the streams, passing frames up to a C# WPF viewer.
- Contract development for ipDatatel
- Wrote uDownloader (C#)
  - Provides remote access to alarm panels
  - Tray application to monitor connections, view data transmission
- Made custom version of com0com null-modem emulator driver (C++)

Adjunct Assistant Professor   2014 - 2016
Mathematics Department
Franklin & Marshall College   Lancaster, PA

- 6 sections of Calculus.
- 3 sections of Algebra & Trigonometry.

Senior Software Engineer   2010 - 2011
Wall Street On Demand   Boulder, CO

- Made an attribute-based runtime object to/from stream protocol mapping engine using compiled expression trees in C#.
- Heavily optimized the mapping of built-in types to/from the stream protocol achieving a 10x speed increase.
- Optimized the memory profile of the multithreaded socket server by writing a lockless thread-safe memory pool in C#. This allowed the server to handle a 2x increase in load.

Senior Software Engineer   2009 - 2010
Synaptics   Santa Clara, CA

- Debugged and fixed issues in the Synaptics TouchPad driver, ranging from blue screens to gesture recognition problems (C++).
- Built a tool to play back and analyze gestures from customer log files (C#).
- Built a tool to verify and optimize driver INF files (C#).

Software Engineer   2007 - 2009
Wall Street On Demand   Boulder, CO

- Rewrote the company's entire distributed computing layer in C#. This included the implementation of socket based clients and servers that communicate using the company's custom data transfer protocol.
- Wrote large portions of Goldman Sachs' (the parent company) new C# financial plotting tool.

- Built the foundation for the company's new chart and pdf generation server. This uses WPF to generate thumbnail charts for web pages as well as creating complete stock and research reports.

Scientific Programmer   2006 - 2007
L-3 Communications - Applied Technologies Division   Santa Barbara, CA
Security Clearance: Secret

- Developed a C# control encapsulating NASA's WorldWind 3D mapping application. Wrote algorithms to display shaded contours on WorldWind's globe.
- Exposed Unmanaged C++ algorithms and graphics packages for use in Managed C# code using both P/Invoke services and C++/CLI wrappers.
- Wrote a C++ application that provides nuclear effects simulation in real-time wargaming exercises for White Sands Missile Range.
- Wrote a COM object that provides source region electromagnetic pulse effects simulation (the thing that happens when a nuke goes off at low altitude).
- Wrote a parallel Monte Carlo electromagnetic simulation to compute damage probabilities.
- Wrote a finite difference time domain Maxwell's equations visualizer in C++.

Education

Ph.D., Mathematics   2011 - 2013
Arizona State University
Research: Discrete math, graph coloring, games and algorithms.
Advisor: Hal Kierstead

M.A., Mathematics   2003 - 2005
University of California, Santa Barbara

B.A., Mathematics   1999 - 2003
Washington University in St. Louis
Ross Middlemiss Prize (for top graduating Mathematics major)
Study Abroad in the Netherlands (Utrecht University)   2001 - 2002

Honors & Activities

Erdős number 2   2011
1st place, Mentor Graphics State Programming Competition   1997 and 1998
Developed Betsy, a master strength chess program, in C   1998 - 2003
Built Tesla coils and produced massive lightning bolts   1997 - 1999

Technical Skills

- **Languages and Technologies**
  - C#, C/C++, JavaScript, Go, Python, Solidity, F#, Dart, Java, Pascal, VB.NET, Scheme, Fortran, x86 assembly
  - LINQ, LaTeX, SQL, MVC, HTML, CSS, XML, jQuery, AJAX, XSLT, Bridge.NET, Google protobufs
  - SQL Server, MySql, Sqlite, Amazon DynamoDB, Azure SQL, Entity Framework
  - WinForms, WPF, Silverlight, Xamarin

- **Development Tools and Platforms**
  - ANTS profiler, dotTrace, NUnit
  - .NET Reflector, dotPeek
  - Visual Studio, Notepad++
  - Git, Subversion, CVS
  - Windows, Linux, Mac

Research Articles

30^+ publications in top-tier discrete mathematics and philosophy journals. A couple of favorites:

D.W. Cranston and L. Rabern. Planar graphs are $\frac{9}{2}$-colorable *Journal of Combinatorial Theory*, 2017.

> This article is about coloring countries on a map so that adjacent countries receive distinct colors. It was conjectured in 1852 that any map could be colored thusly using only 4 colors. This was finally proved in 1976, but the proof is not human-checkable; it requires many hours of computer time to check thousands of cases. Finding a human-checkable proof is still an open problem. To prove that 5 colors suffice is relatively simple. We gave a human-checkable proof that 4.5 colors suffice; this means that we get to use 9 colors, but have to assign each country 2 colors.
>
> - settled a 20 year old conjecture on the existence of such a proof
> - featured on the popular computer science blog Computational Complexity by Lance Fortnow and Bill Gasarch.

B. Rabern and L. Rabern. A simple solution to the hardest logic puzzle ever. *Analysis*, **68**(2), April 2008.

> Three gods A, B, and C are called, in no particular order, True, False, and Random. True always speaks truly, False always speaks falsely, but whether Random speaks truly or falsely is a completely random matter. Your task is to determine the identities of A, B, and C by asking three yes-no questions; each question must be put to exactly one god. The gods understand English, but will answer all questions in their own language, in which the words for yes and no are da and ja, in some order. You do not know which word means which.
>
> - showed how to trivialize the puzzle by asking questions that elicit meaningful answers from Random
> - showed how to solve the puzzle in only two questions by using paradoxes to explode god-heads
> - this article led to the problem getting a lot of press and many follow-up papers have been written (17 so far)