Ömer Sinan Ağacan, CV

Personal info

I'm a Haskell developer with 4 years of experience. I'm currently a Haskell consultant at Well-Typed, where I'm spending most of my time working on the GHC runtime system.

Github: https://github.com/osa1
Blog: http://osa1.net

Education

- Indiana University - Bloomington, Indiana, USA.
  Masters in Computer Science. 2015 - 2016.

- TOBB University of Economics and Technology - Ankara, Turkey.
  BSc in Computer Engineering. 2009 - 2014.

Experience

- **Well-Typed, Haskell consultant. 2018 - ongoing:**
  I'm working on GHC RTS (fixing bugs, implementing a new garbage collector) and various Haskell projects (ranging from debugging and bug fixing to implementing features).

- **Picus Security, Ankara, Turkey. Software Engineer (Haskell). 2017 - 2018:**
  Started building a new version of the product from scratch. Implemented most of the server and peer programs. Designed and implemented engine-to-peer wire protocol, a REST API for querying engine state, a websocket-based protocol for providing live updates to UIs, and other features. Shipped 30 kloc Haskell.

- **Microsoft Research, Cambridge. Research Intern. 2016 Summer:**
  I worked with Simon Peyton Jones on various GHC projects. Two of the bigger projects are: designing and implementing a new intermediate language for GHC (to replace Core), and implementing a new representation for sum types. The latter is merged and available to users with `-XUnboxedSums` flag in GHC 8.2.1.

- **Indiana University, Master of Science in Computer Science. 2015 - 2016:**
  I started as a PhD student, but switched to masters after realizing that I don't want to stay in academia and I enjoy programming more than writing papers. I worked with programming languages team as a research assistant for two years, and I was involved in two published and a few unpublished papers. My research was about optimizing Haskell programs via various high-level program transformations.

- **Runtime Verification. 2014, 4 months:**
  I worked on K Framework as a software engineer. I maintained ktest, a testing framework specifically crafted for K Framework's needs, and worked on the implementation of a new intermediate language.

- **Google Summer of Code for Haskell. 2014, 3 months:**
  I implemented GHC's profiling features (cost-centres, emulated call stacks) for GHCJS, a JavaScript backend for GHC.

- **Soostone - 2014, 3 months:**
  I worked as Haskell software engineer.

- **Formal Systems Laboratory, University of Illinois at Urbana-Champaign. 2013, 3.5-month internship:**
  I worked on K Framework, a rewrite-based executable semantics framework for programming languages, type systems and formal analysis tools. I was advised by Prof. Grigore Rosu.

- **Ozyegin University, Istanbul, Turkey. 2013, 3.5-month internship:**
  Under supervision of Prof. Baris Aktemur: Developed an interpreter for a statically typed multi-staged programming language with subtyping, row polymorphism and type inference. Performed extensive literature reading. Implemented in OCaml.

- **OBSS, Istanbul, Turkey. 2012, 3.5-month internship:**
  Developed a simple static analysis tool for GrayMound, a Java framework based on J2EE and an Eclipse plugin.

- **TUBITAK (The Scientific and Technological Research Council of Turkey), Kocaeli, Turkey. 2010, 4-week internship:**
  Developed the web front-end for PiSi, the package manager of Pardus (a GNU/Linux distribution).

Open Source Contributions

- GHC: I'm a part of The Glasgow Haskell Team. I'm regularly fixing bugs, implementing features, and doing maintenance on the code base.

- GHCJS: I worked on GHCJS during Google Summer of Code 2014. I implemented GHC's profiling features for GHCJS.

- K Framework: During my internship at Formal Systems Laboratory, in addition to fixing bugs and doing maintenance work, I designed and implemented ktest tool. ktest is a program specifically designed for testing K definitions.

- More: I have a long history of open-source contributions. Please see my Github account for more.

Publications

- Ryan R. Newton, Ömer S. Ağacan, Sam Tobin-Hochstadt, Peter Fogg. Parallel Type-checking with Haskell using Saturating LVars and Stream Generators. In the 21st annual ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming. March 2016.

- Edward Z. Yang, Giovanni Campagna, Ömer S. Ağacan, Ahmed El-Hassany, Abhishek Kulkarni, Ryan Newton. Efficient communication and Collection with Compact Normal Forms. In Proceedings of the 20th ACM SIGPLAN International Conference on Functional Programming. September 2015.

- Michael D. Adams and Ömer S. Ağacan. Indentation-sensitive parsing for Parsec. In Proceedings of the 2014 ACM SIGPLAN Symposium on Haskell, Haskell '14, pages 121–132. ACM, New York, NY, USA, September 2014. ISBN 978-1-4503-3041-1. doi: 10.1145/2633357.2633369.

Talks

- Rebooting Supercompilation for Haskell. Haskell Implementors Workshop 2015.