

Árpád Goretity

Software Architect, Data Scientist

h2co3@h2co3.org

github.com/H2C03

linkedin.com/in/h2co3



Education

- **Ph.D. in Computer Science** — 2020–2024 (*in progress*)
Pázmány Péter Catholic University, Faculty of Information Technology and Bionics
Research Topic: Design and Implementation of a Strongly-Typed Data Abstraction Layer and Domain-Specific Language
Supervisor: István Reguly, Ph.D., Tenured Professor
- **M.Sc. in Data Science** — 2018–2020
Università degli Studi di Padova, Dipartimento di Matematica “Tullio Levi-Civita”
Thesis: Towards Personalized Disease Risk Prediction from Metagenome Analysis of the Microbiome
Qualification: 110 / 110, Cum Laude
- **B.Sc. in Molecular Bionic Engineering** — 2013–2016
Pázmány Péter Catholic University, Faculty of Information Technology and Bionics
Thesis: Design and FPGA Implementation of a Protein Structure Comparison Method Based on Alignment of Backbone Conformations
Qualification: Excellent

Programming Experience

- C, C++, Objective-C — 11 years, *highly proficient*
- Rust — 5 years, *highly proficient*
- Python — 6 years, *proficient*
- Swift — 4 years, *intermediate*
- Some exposure to Haskell, C#, R, and VHDL
- Some exposure to various SQL dialects (SQLite, PostgreSQL, MySQL)

Publications

- Árpád, GORETITY; Zoltán, NAGY; Zoltán, GÁSPÁRI;
“Acceleration of a Protein Structure Comparison Algorithm on FPGA,”
in *2017 European Conference on Circuit Theory and Design (ECCTD)*, 2017 [online].
DOI: 10.1109/ECCTD.2017.8093342
- Árpád, GORETITY; István, REGULY;
“Query Complexity in Modern Database DSLs,”
draft, in progress

References – Personal and Professional Projects

- **Metagenomic Pipeline @ Medipredict, Ltd.** (*Python*) — A comprehensive data pre-processing, analysis, modelling, and visualization solution for the discovery of Inflammatory Bowel Diseases, using human metagenomic samples and simple machine learning models. Created in collaboration with experts in biology and bioinformatics.
- **Sparkling** (*C*) — A modern, lightweight, extensible, embeddable scripting language. Compiler, bytecode interpreter, and runtime library implemented from scratch.
- **Avocado** (*Rust*) — A strongly-typed, schemaful, high-performance client library for MongoDB, suitable for Domain-Driven Design.
- **iCsekk @ SciApps.io, Díjbeszedő Holding Zrt.** (*Objective-C*) — iCsekk by Magyar Posta Zrt. is a service for paying utility bills in Hungary. As a contractor, I have completely rewritten the core communication layer of the iOS app, fixing several high-severity security vulnerabilities in the process.
- **CryptTalk @ Arenim Technologies, AB.** (*Objective-C*) — A very high-reliability, end-to-end encrypted VoIP solution for businesses and persons for whom secure calls are mission-critical. Contributed security hardening features and UI improvements to the iOS app.

Technical Talks, Presentations, and Public Lectures

- **Compiler Design and Implementation**
Swift Meetup Budapest, September 2015–June 2016
A 6-part, practice-oriented series, in which participants implement a simple, working compiler in Swift for a procedural toy programming language.
- **Non-Pessimizations in the Swift Compiler**
Swift London, 15 November 2016
A discussion of interesting canonicalization and optimization patterns, as implemented by the Swift compiler.
- **Acceleration of a Protein Structural Alignment Method with FPGAs**
Budapest Science Meetup, 20 February 2017
A simplified recount of my Bachelor's thesis, discussing the significance of hardware acceleration in bioinformatics and big data.

Other Skills

- Languages: **English** (Cambridge C2), **French** (C1), **Italian** (B1), **Hungarian** (native)
- Docker basics and Unix shell scripting
- Source Code Management using Git
- Technical typesetting with \LaTeX
- Teaching experience (TA, later Grad Student Instructor @ PPCU FITB):
 - Linear Algebra I-II — 2014/15/1–2
 - Nonlinear Dynamical Systems in Biology — 2020/21/1
 - Introduction to Programming I — 2015/16/1, 2017/18/1
 - Introduction to Programming II — 2020/21/2
 - Programming I — 2014/15/2
 - Parallel Programming — 2020/21/2