Igor Taraymovich

+7 (916) 729 07 14
tarajmovich.i@phystech.edu
https://github.com/trmigor

Key skills

- Programming languages: C++, Golang, C
- Working experience with Git, AWS, Docker, MongoDB
- Multithreading, software design and engineering, distributed systems, continuous integration, automated testing
- Project management, mentoring, technical training
- Quantitative skills: computational maths, optimisation, algorithms

Education

Moscow Institute of Physics and Technology *September 2017 – Present*
Department of Control and Applied Mathematics

- Bachelor of Applied Mathematics and Physics
- Main specialisation: computer science

Work and research experience

Research assistant *September 2019 – Present*
Moscow Institute of Physics and Technology & Acronis
Moscow, Russia

- Working on research project for study of microservice API automated testing. It is focused on fuzzing and mutation method.
- Aim is to create a tool that processes RESTful API description (in RAML or Swagger), then automatically generates and sends sequences of HTTP requests to check whether API is working in accordance with description.
- Designed and developed data generator to test «dangerous» input values. It creates random values by criteria received from Swagger decoder to check boundary, invalid and special cases.
- Designed and developed random expectedly valid data generator with training opportunity. It uses standard random integer generator to create values according to feedback sent by user.

Projects

Command-line interpreter (Microshell)
https://github.com/trmigor/MicroShell

- Microshell provides most of commands, contained in bash.
- Syntax is the same as in other implementations of UNIX shell, with regular expressions, pipes, input-output redirection and standard notation available.
- Microshell interacts with operating system by a set of UNIX system calls and supports signals and multiprocessing.

Website for programming contests (Judex)
https://github.com/trmigor/Judex

- Judex is a system for automatic testing for student contests.
- Backend is written fully in Golang, using the standard libraries and MongoDB Server. Frontend is using JavaScript to interact with users.
- Project is using multithreading with goroutines for faster results. System calls are used to manage time and memory used for test runs in isolated environment in order to keep system safe.

Languages

English (fluent), French (pre-intermediate), Russian (native), Ukrainian (native)