Przemysław Leśniak

przemek.lesniak1@gmail.com

Education

- **University of Wrocław** — Wrocław, Poland
  B.Sc. currently pursing M.Sc., Computer Science — *Mar. 2013 - Jun. 2019*
  - Bachelor thesis: *virtual memory subsystem for mimiker operating system*
- **Saarland University** — Saarbrucken, Germany
  M.Sc. Computer Science — *Sep. 2017 - Mar. 2018*
  - One semester Erasmus student exchange

Work Experience

- **Bloomberg** — London, United Kingdom
  Software Engineering Placement — *Mar. 2018 - Sep. 2018*
  - Joined Market Data Pipeline team working on system publishing market data to interested subscribers.
  - Responsibilities: mdpconf project that centralized configuration of complicated pipeline that in future would make it easier to improve scalability of the system, tools used by the team for debugging and inspecting state, configuring the pipeline.
  - Technologies used: C++, python2, javascript, comdb2, docker.
  - Won *Best Dancing Intern* award.
- **Saarland University** — Saarbrucken, Germany
  Student Programmer — *Nov. 2017 - Mar. 2018*
  - Extended x86 backend in LLVM to support Intel MPX instructions. Used in research project to improve memory safety of C programs by inserting runtime checks in compiled code.
  - Technologies used: C++, LLVM
- **Google Summer of Code** — Remote Work
  Student Programmer — *June. 2017 - Sep. 2017*
  - Improved vectorization by adding metadata and fixing bugs in LLVM IR generation which resulted in performance of executed code by 400% in some common cases.
  - Technologies used: C++, LLVM, Python
- **Nokia** — Wrocław, Poland
  Compiler Developer — *Jul. 2016 - Mar. 2017*
  - Joined team responsible for TTCN-3 language compiler and runtime used at Nokia for writing and running complicated test scenarios.
  - Greatly reduced number of memory allocations in runtime by using object-pool like design pattern resulting in 20% performance gain on average.
  - Reduced number of copying operations by introducing move operation and generating it in compiler which resulted in 5-10% performance gains on average.
  - Technologies used: C++, Python, TTCN-3, valgrind

- **Nokia** Wrocław, Poland
  Summer Intern *July. 2015 - Sep. 2015*
  - Participated in library design inspired by Parsec library from Haskell language that was used to implement partial parser for TTCN-3 language.
  - Designed and implemented algorithm based on pushdown automata to locate changes in code in real time that would need to be re-parsed.
  - Integrated the algorithm and the parser into QtCreator to provide IDE functionality like auto-completion and jumping to function definitions.
  - Technologies used: C++, Haskell, SQLite

Projects

- **mimiker**
  University of Wrocław operating system
  - Significant contributions in kernel: physical page allocator, page table management, tlb refill handler, mutexes, basic in-memory file system, ramdisk loading, selected system calls. Helped new students get into the project.
- **quant**
  Lossy image compression algorithm
  - Reduces image size by approximately 80% using vector quantization method. Preserves good image quality. Optimized typically slow algorithm by fine tuning data structures like using small vector and parallelizing parts of code.
- **C compiler**
  Compiler construction course project
  - Written in C++. Implemented hand written parser, semantic analysis (type checking, scope resolution), code generation to LLVM IR, Static Constant Propagation optimization on LLVM IR. Supports basic programming constructs like if statements, while loops, structs, pointers, string literals.
- **Other projects**
  Written at home and university
  - Tiny compiler for subset of C written in Haskell.
  - Program for graph drawing and running complex graph algorithms written in Java.
  - Rubik's cube solver written in python.
  - Raytracer written on Computer Graphics course in C++.

Skills and interests

Programming languages: C++, C, Python, Java, Javascript, Haskell

Other: git, linux, vim, valgrind, object oriented programming

Interests: popping dance, speedcubing