Михаил Разаков

Compiler Backend / Code Generation Engineer

razakov.mikhail@outlook.com
t.me/Micodiy
github.com/Misha1302

Разрабатываю backend компилятора: проверка SSA/CFG, анализ живости, распределение регистров, spills, phi lowering и генерация SysV x86-64. Проверяю назначения независимым verifier и сравниваю машинное исполнение с интерпретатором; ранее работал с LLVM 22/C++ в МЦСТ.

2022: Компиляторы / VM · 2023: CIL + native x86-64 · 2024–2025: AST / IR / backends → toolchain · 2026 →: LLVM + program analysis

ПРОФЕССИОНАЛЬНЫЙ ОПЫТ

октябрь 2026 — настоящее время

ИСП РАН — разработка SharpChecker

- Работаю над символьными вычислениями в SharpChecker — инструменте статического анализа C#/.NET.

июль — август 2026

МЦСТ — стажёр по разработке компиляторов

- Реализовал LICM-проход для LLVM 22 на C++; перенос выполняется только при выполнении условий по структуре цикла, обращениям к памяти, побочным эффектам и безопасности спекулятивного выполнения.
- Собрал Python-обвязку для сравнения поведения программы до/после оптимизации и проверки случаев с преобразованием и без него.

ПРОЕКТЫ

x86-64 Compiler Backend & Register Allocation

Проверка SSA/CFG → анализ живости и интерференции → register allocation/spills → phi lowering → генерация SysV x86-64.

Verifier раскладки + сравнение native execution с эталонным интерпретатором; актуальный CI проекта зелёный; отдельная публичная линия native-codegen начинается с Visk (2023) и Wist4 (2024).

LLVM Interprocedural Global IV Optimization

LLVM 22 pass локализует поддерживаемые глобальные переменные, изменяющиеся как индукционные, после межпроцедурного анализа эффектов, аффинной эволюции и отдельной проверки условий применимости; APInt моделирует арифметику фиксированной разрядности.

29 позитивных и негативных регрессионных сценариев: LLVM Verifier, структурные ожидания, идемпотентность и сравнение наблюдаемого поведения до/после преобразования.

UniversalToolchain / Wist2

Разрешение зависимостей и конфликтов, выбор реализаций, порядок проходов и типизированные маршруты артефактов до исполнения.

1 324/1 324 теста; interpreter/CIL parity и проверка конфигурации до runtime.

КОМПЕТЕНЦИИ

Backend / Codegen
x86-64, SysV ABI, stack frames, calling conventions, liveness/interference, register allocation, spills, phi lowering, native execution

Low-level / Assembly
Iced.Intel code emission, executable-memory JIT experiments; NASM IA-32, CDECL, x87

LLVM / IR
C++23, LLVM 22/IR, CFG/SSA, анализ циклов, межпроцедурный анализ, преобразования IR

Проверка
verifier раскладки, LLVM Verifier, структурные проверки, идемпотентность, differential execution

ОБРАЗОВАНИЕ

НИУ ВШЭ — Программная инженерия

ДОСТИЖЕНИЯ

Отбор МЦСТ — LLVM / C++
По письму организатора: единственная оценка 5,0/5,0 и единственное решение, прошедшее 104/104 теста, среди 49 присланных работ.

LangDev'26 — Build the Language, Then Make the Abstractions Disappear
9 октября 2026 выступил на LangDev с докладом об архитектуре UniversalToolchain.

ФКН НИУ ВШЭ — конкурс проектов с открытым кодом
UniversalToolchain / Wist2 — один из победителей конкурса проектов с открытым кодом студентов ФКН.

Другие достижения
«Юниор» НИЯУ МИФИ: победитель ИТ-секции в 2025 и диплом I степени в 2026; дипломант «Высшей пробы» по промышленному программированию; Балтийский конкурс: диплом I степени и премия «Совершенство как надежда».

Москва / удалённо