

Стемпоржецкая Людмила

Email:

GitHub

GPA

+7 (909) 432-90-83

✉ stemporzhetskaia.lv@phystech.edu

✉ StemporzheckayaLudmila@yandex.ru

🔗 ludmilastemp

8.4

## ОБРАЗОВАНИЕ

---

**Специализированный учебно-научный центр (факультет) школа-интернат имени А.Н. Колмогорова**

Москва, 2021 - 2023

**Московский физико-технический институт (национальный исследовательский университет)**

Долгопрудный, 2023 - настоящее время

## ПРОЕКТЫ

---

### SPU

<https://github.com/ludmilastemp/CPU>

2023, Октябрь

Стековая виртуальная машина и компилятор. Текстовый файл на языке ассемблер обрабатывается, ищутся лексические ошибки, расставляются метки. Все команды кодируются и получается файл с исполняемым байт кодом.

- Ассемблер/дизассемблер
- Кодировка инструкций, ISA (instruction set architecture)
- Работа с метками (fixups)
- Работа с текстовыми и бинарными файлами

### Akinator

<https://github.com/ludmilastemp/Akinator>

2023, Ноябрь

Игра Акинатор. Суть игры заключается в отгадывании программой загаданного объекта, с помощью простых вопросов (ответы да/нет). В случае, если игра еще не знает такого объекта, предлагается внести его в базу. Также можно запросить описание объекта. Все данные хранятся в двоичном дереве.

- Структура данных двоичное дерево
- Dot, Graphviz

### Wolfram

<https://github.com/ludmilastemp/Wolfram>

2023, Ноябрь

Программа для работы с математическими выражениями. Обработка выражения, его упрощение, а также вычисление первой и второй производных. Все промежуточные результаты с подробным описанием записываются в файл в формате LaTeX.

- Дерево выражений
- Dot, Graphviz
- LaTeX

### Mandelbrot

<https://github.com/ludmilastemp/Mandelbrot>

2024, Март

Отрисовка множества Мандельброта. Исследование векторных инструкций SIMD (расширения AVX, SSE) и ускорение программы с их помощью. Проведено 4 итерации, впоследствии чего получено ускорение в 3.5 раза.

- Векторные инструкции SIMD
- Графическая библиотека txlib
- Linux perf утилита

### HashTable

<https://github.com/ludmilastemp/HashTable>

2024, Апрель

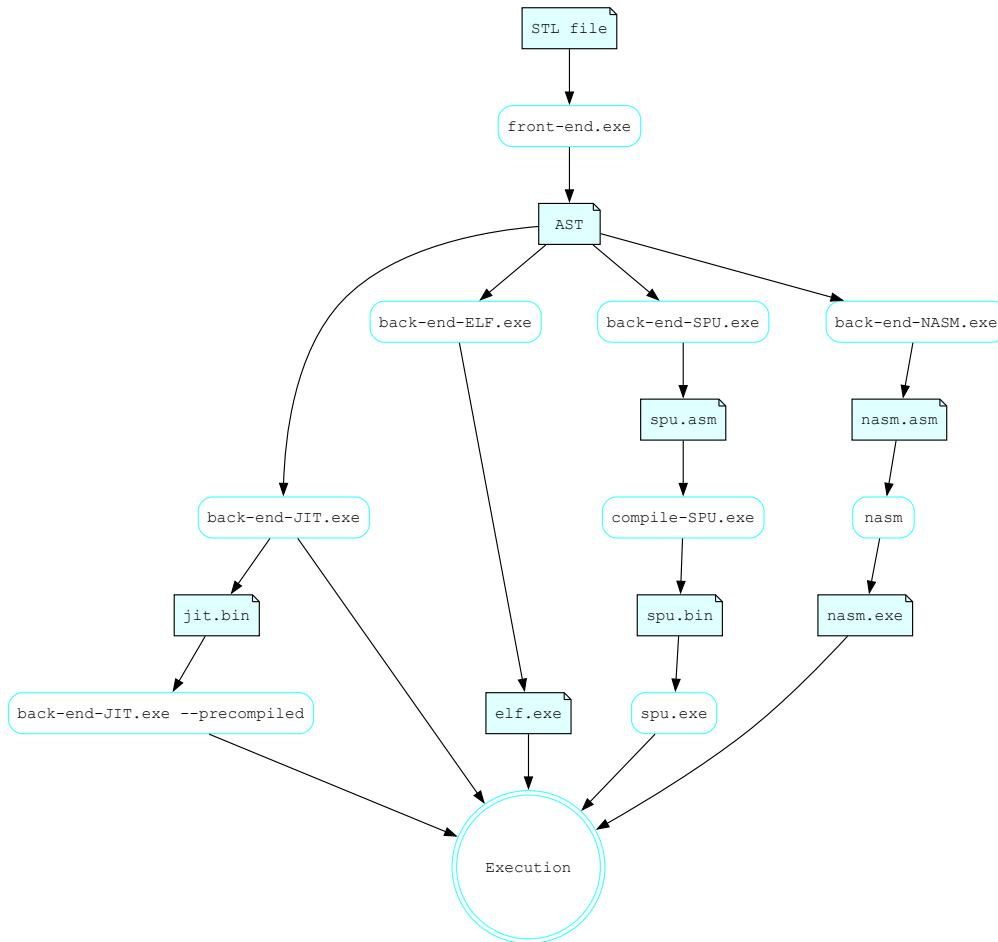
Хеш-таблица (метод цепочек). Исследование различных хеш-функций, их load factor-a. Реализация 3 методов оптимизации: функция на ассемблере, ассемблерная вставка, SIMD инструкции. Общее ускорение составило +300%.

- Структура данных хеш таблица
- Векторные инструкции SIMD
- Inline Assembly
- Linux perf утилита

2023, Декабрь - 2024, Май

Программная инфраструктура языка программирования STL. Front-end: лексер, парсер на основе рекурсивного спуска. 4 различных back-end-а:

- *back-end-ELF* (из AST дерева в исполняемый файл ELF)
- *back-end-JIT* (JIT (Just-In-Time) AST дерева в бинарный файл x86)
- *back-end-NASM* (из AST дерева в файл на языке NASM)
- *back-end-SPU* (из AST дерева в файл на языке для STL-SPU)



- Front-end/Back-end компилятора
- Кодировка инструкций x86
- Структура исполняемого файла ELF
- JIT компиляция

## ОПЫТ РАБОТЫ

**Языки программирования:** C, x86-64 Assembly (NASM), Python

**Инструменты:** perf, git, make, gdb

**Структуры данных:** стек, очередь, бинарное/биномиальное/AVL/splay/B/red-black деревья, дерево отрезков/фенфика, двоичная/биномиальная/фибоначчиева кучи, хеш-таблица

**Языки разметки:** Markdown, dot, LaTeX