

# МОТЫГУЛЛИН БУЛАТ

+79172841606 | [bulatmot@gmail.com](mailto:bulatmot@gmail.com) | [t.me/bulat1337](https://t.me/bulat1337) | [github.com/bulat1337](https://github.com/bulat1337)

## ОБРАЗОВАНИЕ

**МФТИ - Московский физико-технический университет**

Москва

Бакалавр ФРКТ - Физтех-школа радиотехники и компьютерных технологий

2023 - 2027 | 2 курс

**Основные курсы:**

Математический Анализ, Линейная Алгебра, Теория Вероятности, Информатика, Физика

## ПРОЕКТЫ

**Tatlang** | C, Make, Git, CPU emulator, Binary tree, Recursive descent parser

[github.com/bulat1337/Tatlang](https://github.com/bulat1337/Tatlang)

- Написал язык программирования. Синтаксис адаптирован под татарский язык и поддерживает все символы Unicode таблицы.
- Разработал компилятор. Основные этапы работы: токенизация, формирование абстрактного синтаксического дерева (АСД) и ассемблирование.
- Адаптировал процесс ассемблирования под эмулятор процессора.

**Ray Tracing** | C++, CMake, Git

[github.com/bulat1337/Ray\\_Tracing](https://github.com/bulat1337/Ray_Tracing)

- Разработал движок трассировки лучей, который вручную рассчитывает путь каждого луча в 3D-сцене.
- Реализовал механизмы сглаживания, глубины диффузии и эффекта размытия, что позволяет пользователям настраивать баланс между качеством изображения и производительностью.
- Оптимизировал процесс рендеринга с помощью структуры BVH (Bounding Volume Hierarchy) для ускорения проверки пересечений лучей с объектами.

**CPU** | C, Make, Git, Stack, SFML

[github.com/bulat1337/CPU](https://github.com/bulat1337/CPU)

- Эмулировал работу процессора. Основные этапы работы процессора: формирование байт-кода и его исполнение.
- Разработал язык ассемблера.
- Добавил поддержку арифметических операций, ввода и вывод данных. Реализовал работу с регистрами, стеком и оперативной памятью.
- Эмулировал видеопамять. Для формирования изображения использовал графическую библиотеку SFML.

**Differentiator** | C, Binary tree, Recursive descent parser, Graphviz

[github.com/bulat1337/Differentiator](https://github.com/bulat1337/Differentiator)

- Разработал программу для дифференцирования и упрощения математических выражений.
- Добавил генерацию описания процесса работы в формате Tex.
- Выражения представлены в формате бинарного дерева. Для графического представления деревьев использовал Graphviz.
- Реализовал анализ математических выражений с помощью алгоритма рекурсивного спуска.

**Akinator** | C, Make, Git, Binary Tree, Graphviz

[github.com/bulat1337/Akinator](https://github.com/bulat1337/Akinator)

- Написал игру, которая угадывает загаданный объект, задавая пользователю вопросы

**Range Queries** | C++, CMake, Git, Red-black Tree, Graphviz

[https://github.com/bulat1337/Range\\_Queries](https://github.com/bulat1337/Range_Queries)

- Создал интерактивную программу на основе своего черно-красного дерева, позволяющую добавлять узлы и запрашивать их количество в заданных границах.

**Triangle Intersection** | C++, CMake, Git, Spacial Hashing

[https://github.com/bulat1337/Triangle\\_Intersection](https://github.com/bulat1337/Triangle_Intersection)

- Реализовал алгоритм, проверяющий пересечение треугольников в трехмерном пространстве.
- Оптимизировал алгоритм для обработки пересечений большого количества треугольников, применив пространственное хеширование (Spatial Hashing).

**Adaptive Replacement Cache** | C++, CMake, Git

<https://github.com/bulat1337/ARC>

- Реализовал адаптивное кэширование данных
- Написал "идеальный" алгоритм кэширования, для сравнения результатов.

## О СЕБЕ

Учась в школе, я участвовал в олимпиадах по физике, стал призёром в:

- Инженерная олимпиады школьников (физика) - призёр
- Олимпиада Курчатов физика - призёр

На физтехе я получил навык работы в команде, научился грамотному техническому общению. Опыт мозгового штурма и совместного решения сложных задач поможет мне работать в профессиональной команде, прислушиваться к каждому и эффективно находить лучшее решение проблемы.