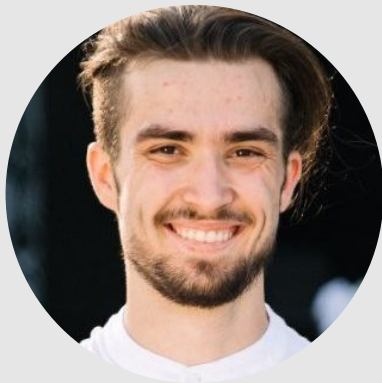


# Данил Герасименко

студент МФТИ



## Обо мне

Я студент 3го курса МФТИ факультета прикладной информатики и математики (МФТИ ФПМИ)

## Навыки общения и ментальность

Дипломатичный

Креативный

Любознательный

Быстро адаптируюсь

Имею работать в команде

Терпеливый

Стрессоустойчивый

Чуткий

## Профессиональные навыки

Языки программирования:

C (имею достаточный опыт)

Assembler (имею достаточный опыт)

C++ (имею достаточный опыт)

Python (вспомогательные задачи)

Графические библиотеки:

Vulkan

SDL 2

Unity (изучил учебный курс)

Unreal Engine (изучил учебный курс)

Другие языки:

Markdown

Latex

Bash (скрипты для тестирования)

Остальные навыки:

Git

Cmake or Make

Linux (знаком с shell оболочкой Linux)

## ОБРАЗОВАНИЕ

2021-В  
процессе

**Бакалавриат  
ФПМИ МФТИ**

📍 Долгопрудный

Окончил следующие курсы:

- Математический анализ
- Линейная алгебра
- Дифференциальные уравнения
- Комбинаторика и теория чисел
- Математическая логика
- Математическая статистика
- Теория вероятности и меры
- Общая физика
- Аналитическая механика
- Теория поля
- Операционные системы
- Основы машинного обучения

2021-2022

**Компиляторные технологии и  
индустриальное программирование**

📍 Долгопрудный  
Дединский Илья

Дополнительный курс C от Дединского Ильи.

2022-2023

**Использование C++**

📍 Долгопрудный

Владимиров К.И.

Дополнительный курс C++ от Владимира К.И.

2022-2023

**Основы операционных систем** 📍 Долгопрудный  
Луканин А.А.

Дополнительный курс, посвященный изучению инструментов языка C для работы внутри ядра системы UNIX.

2023-2024

**Математические основы  
визуализации**

📍 Долгопрудный

Афанасьев В.О.

Дополнительный курс о математической стороне разработки графических движков

## ОПЫТ РАБОТЫ

07.23-09.23

**Стажировка в МЦСТ**

📍 Москва

МЦСТ

Во время стажировки в МЦСТ я занимался разработкой бинарного компилятора. Первой моей задачей было ускорение модуля программы с помощью ассемблерных вставок. Я реализовал векторные операции над built-in типами. Результатом работы стало повышение КПД в соответствующих модулях. Вторая задача была связана с генерацией текста ассемблерного кода для тестирования разработанных модулей из первой задачи. Выполнение этой задачи позволило нам отладить общее выполнение бинарного компилятора на краевых тестах.

10.23-В  
процессе

**Сотрудник ИСП РАН**

📍 Москва

ИСП РАН

На данный момент работаю в в отделе рендеринга больших объектов в должности лаборанта-студента. Основная задача — разработка графического движка на языке C++ с использованием библиотеки API Vulkan от Khronos Group. Модули движка спроектированы с учетом принципов ООП. На данный момент написан модифицированный конвейер для рендеринга и отрисовки текстур (в том числе прозрачных) с большим количеством точек при диффузном освещении. В планах — реализация технологии трассировки лучей и расчет специальной геометрии на вычислительных шейдерах.

## Контакты

📞 BileyHarryCopter

✉ gerasimenko.dv@phystech.edu

✉ +7(991)-082-13-50

+995 (555)-383-542

📍 Фридона Халваши, 20, Батуми

## Языки

🇷🇺 Русский - родной

🇬🇧 Английский - B2-B1

## 🔗 ПРОЕКТЫ

### Assembler:

🔗 Разработка простого бинарного компилятора как задание по стажировке в МЦСТ. Я реализовал несколько вычислительных и алгоритмических оптимизаций, которые улучшили КПД ассемблерных модулей бинарного компилятора.

### C:

🔗 Реализация архитектуры стека с использованием кодирования в стиле #define

🔗 Akinator - реализация известной игры с самообучающимся бинарным деревом

🔗 Differentiator - реализация дифференцирующей машины с использованием бинарного дерева и рекурсивного спуска

🔗 Реализация последовательного контейнера в виде списка и визуализация его работы с помощью GraphViz.

🔗 Hashmap (собственная копия std::map)

🔗 LFUDA - реализация кэширования по политике LFUDA (комбинация лучшего из LFU и LRU). Групповой проект, в котором я занимался тестированием, отладкой и разработкой внутренних компонентов.

🔗 Operation\_System — репозиторий с исходным кодом для решения конкурсных задач, связанных с операционными системами на UNIX. Здесь рассматривается: взаимодействие процессов друг с другом, структура и организация файловой системы, а также обработка сигнальных вызовов в режиме ядра.

### C++:

🔗 Реализация кэша LFUDA и Belady. В данном проекте ставилась задача сравнить скорость выполнения кэшей, написанных на разных языках: C и C++. Также в ходе выполнения данной задачи алгоритм Беладди был представлен как эталон кэширования, по сравнению с которым политика LFUDA показала хорошие результаты.

🔗 Класс матриц RAII. Здесь я реализовал пользовательский класс массива (он же std::array) и на его основе был реализован класс матрицы.

🔗 SDL2 Tutorial, по которому я понял механику данной графической библиотеки. С помощью SDL2 была разработана следующая игра.

🔗 Backgammon - реализация простого 2D игрового движка и соответственно реализация известной игры Backgammon (она же Нарды). Внутренняя реализация программы выполнена на основе ООП. Алгоритм игры с компьютером разработан на основе конечного автомата.

🔗 HW3D (Обнаружение пересечений треугольников и их визуализация с помощью vulkan api). Групповой проект, в котором я занимался разработкой примитивов геометрии, алгоритмом пересечения 2D и отладкой. В ходе этого проекта нам удалось создать программу, которая может довольно быстро вычислять пересечение до 1 000 000 треугольников.

🔗 vulkan\_dev - реализация графического движка с использованием Vulkan Api от Khronos Group. Графический движок создан на основе ООП и на данный момент способен обрабатывать миллионы точек с диффузным освещением и прозрачными текстурами при стабильном значении fps.