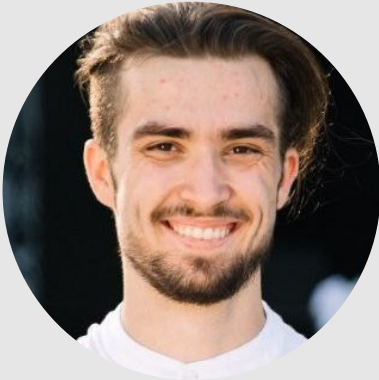


# Danil Gerasimenko

MIPT student



## About me

I'm a 3rd grade student of MIPT DIHT (Department of Innovation and High Technology).

## Strengths and Soft Skills

Diplomacy

Creativity

Curiosity

Adaptability

Team Working

Patience

Stress-resistant

Sensible

## Professional Skills

Programming Languages:

C(good practise)

Assembler(good practise)

C++(not bat practise)

Python(auxiliary aims)

Graphical API:

Vulkan

SDL 2

GraphViz

Unity (student course)

Unreal Engine (student course)

Other Languages:

Markdown

Latex

Bash (using scripts in testing)

Application Skills:

Git

Cmake or Make

Linux (familiar with Linux shell)

## EDUCATION

2021-  
Ongoing

### Bachelor Degree

MIPT DIHT

Finished courses:

- Linear Algebra;
- Probability Theory
- Combinatorics
- Math statistics
- Math Analysis
- Differential Equations
- General Physics
- Analytical mechanics
- Field theory
- Operation Systems
- Microcontrollers
- Basic of machine learnings

📍 Dolgoprudniy

2021-2022

### Compiler technology and professional programming

Ilya Dedinsky

Auxiliary course of C by Ilya Dedinsky.

📍 Dolgoprudniy

2022-2023

### Uses and Applications of C++

Vladimirov K.I.

Auxiliary course of C++ by Vladimirov K.I.

📍 Dolgoprudniy

2022-2023

### Operating Systems Basics

Lukanin A.A.

An additional course devoted to the study of C language tools for working inside UNIX operating systems.

📍 Dolgoprudniy

2023-2024

### Mathematical basics of visualisation

Afanasiev V.O.

Auxiliary course about mathematical side of graphical engines' development

📍 Dolgoprudniy

## WORK EXPERIENCE

July -  
September  
2023

### Internship in MCST

MCST

During this internship in MCST, I was involved in the development of a binary compiler. My first task was to speed up a program block with assembler inserts. I implemented vector operations on built-in types. The result of the work was an increase KPI in the corresponding modules. The second task was related to the text generation of assembler code for testing the developed modules from the first task. Completing this task allowed us to debug the overall execution of the binary compiler on edge tests.

📍 Moscow

October  
2023 -  
Ongoing

### Work in ISP RAS

ISP RAS


At the moment I work at ISP RAS in the department of rendering large-scale objects as a student lab assistant. My main task is developing a graphics engine based on C++ using the Vulkan API library from Khronos Group. The engine modules are designed taking into account all OOP principles. At the moment I have written a modified pipeline for rendering and drawing textures (including transparent ones) with a large number of points under diffuse shading. The plans include implementing ray-tracing technology and calculating special geometry on computational shaders.

📍 Moscow

## Contacts

 BileyHarryCopter


 gerasimenko.dv@phystech.edu


 +7(991)-082-13-50


+995 (555)-383-542

 Pridona Chavashi, 20, Batumi

## Languages


 Russian - Native

 Ukrainian - Dialect

 English - B1-B2


## PROJECTS


### Assembler project:


 Development of simple binary compiler as task from internship in MCST. I implemented several calculating and algorithmic optimizations which improved KPI of assembler modules of binary compiler.

### C projects:


 Realisation of stack architecture using #define style coding.


 Implementation of famous game Akinator. Console interface. Inner architecture is realized using binary tree.

 Differentiator - implementation of differentiation machine using binary tree and recursive descent technology (elementary parser)


 Realisation of sequential container as list and visualisation it's working using GraphViz


 Hashmap (aka copy std::map)


 LFUDA - implementation of cache by LFUDA politics (combination of the best of LFU and LRU). Group project, in which I was engaged in testing, debugging and developing of inner components.


 Operation\_System is a repository with source code for solving contest problems related to the operating systems on UNIX. The information is covered here: the interaction of processes with each other, the structure and organization of the file system, as well as the processing of signaling calls in kernel mode.


### C++ projects:


 LFUDA and Belady cache implementation. This task was intended to compare the execution speed of caches written in different languages: C and C++. Also, during this task, the Beladi algorithm was presented as a benchmark caching, in comparison with which the policy LFUDA gave good results.

 RAII class of Matrices. In this case I implemented custom class of array (aka std::array) and matrix class was realised on this base.

 SDL2 Tutorial, for which I understood the mechanics of this graphic library. With the help of SDL2 I developed the next game.

 Backgammon - implementation of simple 2D game engine and respectively realization of famous game backgammon (aka Nardy). The internal implementation of the program was done on the basis of OOP. The algorithm for playing with a computer was developed on the basis of a finite state machine.

 HW3D (Triangle collision detection and their visualisation using vulkan api). Group project, in which I was involved in developing the geometry primitives, the 2D intersection algorithm and debugging. During this project, I managed to create a program that can fairly quickly calculate the intersection of up to 1,000,000 triangles.

 vulkan\_dev - implementation of graphical engine using Vulkan Api by Khronos Group. The graphical engine was created on the basis of OOP and at the moment it is capable of processing millions of points with diffuse lighting and transparent textures at a stable fps value.