Vuk Lužanin

Belgrade, Serbia — luzaninvuk@gmail.com — **Github:** https://github.com/Vuk-Luzanin — **LinkedIn:** Vuk Lužanin

RESEARCH INTERESTS

- Computer architecture, RISC-V and AI accelerators, and hardware–software co-design.
- Parallel systems and low-level kernel performance for multiprocessor and AI workloads.
- High-performance computing, scalability, and reliability in multiprocessor environments.

EDUCATION

Fourth year at School of Electrical Engineering, University of Belgrade Belgrade, Serbia
B. S. Software Engineering, GPA: 9.54 / 10 Oct 2022 — Present
Relevant Coursework: Operating systems, Computer Architecture, Algorithms and Data Structures, OOP, Databases

Gymnasium Čačak Čačak, Serbia
Regional center for Talent Youth (field: Mathematics) Sep 2018 — June 2022
Completed Gymnasium in Čačak according to IOP3 standard (field: Mathematics), awarded the "Vuk Karadžić" diploma for outstanding academic achievement
Attended Petnica Science Center (Physics department)

RESEARCH EXPERIENCE

CENTER FOR DATA PROCESSING, School of Electrical Engineering, University of Belgrade Serbia
Research Student Apr 2025 — Present
Parallel Feynman–Kac Algorithm: Performance Comparison and Real-World Applications
Implemented parallel CPU and GPU versions of the Feynman-Kac algorithm using OpenMP, pthreads, C++ threads, SYCL, and CUDA, including applications of the algorithm in practical scenarios. Performed performance analysis (speedup, scalability) and automated benchmarking using Python and Makefiles under the supervision of Prof. Marko Mišić, PhD.

TEACHING EXPERIENCE

SCHOOL OF ELECTRICAL ENGINEERING Belgrade, Serbia
Student Teaching Assistant Oct 2024 — Present
Providing support for classroom activities, including tutoring, grading homework, and assessing exams across multiple courses in which I have shown great proficiency (Operating Systems, Computer Architecture, Algorithms and Data Structures, Databases, OOP, Computer Networks...).

INDUSTRY EXPERIENCE

TENSTORRENT INC. Belgrade, Serbia
Software Engineer Intern – AI Performance & Debugging Tools Sept 2025 — Nov 2025
Working on a custom debugger for Tenstorrent's RISC-V–based AI accelerators as part of the Low Level Kernels, Debug team. I developed a new method for extracting and interpreting debug bus signals, providing clearer visibility into core-level execution state and exposing characteristic hang signatures in multiprocessor workloads. Tech stack: Python, C, RISC-V.

BLACKROCK London and Belgrade
Software Engineer Intern June 2025 — Aug 2025
Designing and implementing a centralized request routing microservice for the Private Markets division, serving as a unified entry point for communication with internal BlackRock services. The service performs endpoint discovery, searches possible routes based on predefined configurations, and forwards requests via gRPC. Tech stack: Spring Boot, gRPC, MSSQL and Liquibase.

ROAD ARROW – Formula Student Team Belgrade, Serbia
Embedded Software Engineer Oct 2024 — Present
Developed and optimized embedded software for vehicle control systems on STM32 microcontrollers, covering real-time data acquisition, sensor integration, and communication protocols (CAN, I2C, UART, SPI). Implemented a CAN-based bootloader for remote code deployment and designed a new modular ECU. Ensured high performance and reliability in a competitive racing setting, helping the team qualify for international competitions in Italy, Poland, and Croatia.

PROJECTS — available on GitHub

RAID Disk Management and Fault Tolerance System
This project integrates RAID support into the xv6 (Unix-like) operating system, enabling configurations such as RAID0, RAID1, RAID0+1, RAID4, and RAID5. It focuses on fault tolerance, data redundancy, and safe parallel access for improved disk reliability and system performance, with optimizations for parallel execution of read and write requests in multiprocessor systems.

FPGA-based Ray Tracing Engine with Custom picoRISC CPU
The project focused on running 3D ray tracing graphics on a Cyclone III FPGA, developed using Quartus 13.1. It involved designing a custom picoRISC CPU based on RISC-V, which executed assembler code with a custom instruction set. The system included a GPU for VGA output, SDRAM communication module (within the GPU), and a PS/2 keyboard controller.

Migration of Student Formula ECU Software to FreeRTOS Architecture
In this project, the existing Engine Control Unit (ECU) software for the student formula car was migrated to use FreeRTOS (a Real-Time Operating System). The new architecture features task-based scheduling, inter-task communication mechanisms (queues, semaphores), and improved system modularity and responsiveness, enabling more reliable and efficient real-time control of engine parameters.

Bootloader for STM32 Microcontrollers
Developed a bootloader system for multiple STM32 microcontrollers on a shared CAN bus using the OpenBLT library. Implemented a firmware update mechanism where a Raspberry Pi fetches precompiled firmware from GitLab and distributes it via CAN protocol, with microcontrollers addressing updates based on their CAN IDs. Designed the system to launch the user application and return to bootloader mode upon receiving a new firmware update request.

Operating System for RISC-V architecture
Implementation of a kernel from scratch subsystem for preemptive, multi-threading for RISC-V CPU with time sharing on a single CPU, supporting both synchronous and asynchronous context switching, event and signal handling and implementation of semaphores using C++ and GNU Assembler.

Autoencoder and Variational Autoencoder Neural Networks for Denoising Microscope Images
Developed autoencoder models for medical and image data processing, aimed at using optical instead of expensive electron microscopes with models for image enhancement. Implemented a convolutional autoencoder for denoising tissue images, an autoencoder for detecting malaria-infected cells via reconstruction error, and a variational autoencoder for MNIST analysis with latent space optimization.

LSM-index for efficient query execution using B+ trees
The project involves implementing a B+ tree-based LSM index in C++, allowing dynamic index creation depending on the queries being executed, over both key and non-key attributes for efficient database management.

AudioStream Management System
The system manages users, audio recordings, and subscriptions with data stored in MySQL databases. Users can upload, categorize, and interact with audio files, including playback, ratings, and favorites. Subscription management ensures dynamic pricing and prevents overlapping subscriptions. Technologies used: JMS, REST, JPA

AWARDS

- Second and Third place in the National Mathematics Competition.
- Third place in the National Physics Competition.
- Third place in the National Mathematics Competition organized by the Center for Talent Youth.
- Scholarship recipient: Scholarships for Exceptionally Gifted Students awarded by the Ministry of Education of the Republic of Serbia.
- Third place at Rivian Computer Vision Hackathon 2025.

SKILLS

Technical: Python, Java, C/C++, SQL, Git, OpenMP, CUDA, Makefile
Languages: English – Advanced (TOEFL iBT 101/120, issued Nov 2025), German – Intermediate, Serbian – Native

Interests: As a computer science student, I'm eager to learn, solve challenging problems, and apply new knowledge. I'm especially interested in multiprocessor systems, computer architecture and high-performance computing. I enjoy working in diverse teams and interacting with people from different backgrounds. Outside of tech, I'm passionate about Karate and Basketball, which bring balance and discipline to my life.