

Konstantinos Despoinidis

Syggrou 4, Stavroupoli, Thessaloniki, Greece | despoinidisk@gmail.com | 6988011379

linkedin.com/in/konstantinos-despoinidis | github.com/KDesp73

Personal Details

Full name: Konstantinos Despoinidis
Father's name: Christos
Date and place of birth: 27/08/2003, Thessaloniki
Nationality: Greek
Military service: Outstanding

Education

International Hellenic University, BSc with an Integrated Master in Information and Electronic Engineering July 2026

- Average: 7.79

2o Geniko Lykeio Kilkis, High School Diploma June 2021

- Average: 19.6

Armoniko Odeio Kilkis, Music Theory (Armonia) January 2020

- Grade: 9.21

Foreign Languages

English, B2 level certificate March 2017

- Certifying body: Pearson
- Certificate number: 601/5625/9

Experience

Lioncode, Software Developer 12/2025 – Present

- Development and maintenance of custom software and solutions to enhance the functionality of the company's products.
- Design and implementation of standalone tools, microservices and RESTful APIs/services for internal use and integration with third-party applications and websites.
- Participation in the design and technical implementation of European R&D projects in collaboration with municipalities (Greece and abroad), academic institutions and private organizations.
- Design and development of applications from greenfield development to their full integration into production.
- Redesign, refactoring and optimization of existing early-stage prototypes to upgrade them to production-ready stage.
- Setup and management of a local cluster (Mac Mini) for hosting and running language models (on-premise LLMs / local inference).
- Development of Retrieval-Augmented Generation (RAG) architecture using Vector Databases and custom harnesses for integrating LLMs into the company's applications.

Volunteering

Scouts of Greece (SEP), Scout Leader (2nd Scout System, Kilkis) 2021–2026

- Design and implementation of educational and recreational programs for children aged 6–11 (cubs).
- Organization, coordination and safe execution of outdoor activities, camps and educational excursions in urban and natural environments.
- Management of the unit's financial resources, preparation of budgets and handling of the required administrative and clerical procedures.
- Regular communication and relationship-building with parents, ensuring children's active participation in activities.

Free and Open Source Software

Irida

KDesp73/irida

- Chess engine compatible with the UCI protocol, written in C as part of the diploma thesis.
- Negamax search algorithm with alpha-beta pruning, enhanced with iterative deepening, quiescence search, transposition tables, null move pruning, late move reductions, aspiration windows, Principal Variation Search (PVS) and MVV-LVA move ordering.
- Position evaluation in centipawns from the side-to-move's perspective: PeSTO-style implementation with discrete material values for the middlegame and endgame, piece-square tables and gradual phase transition (tapered evaluation).
- Additional evaluation heuristics: pawn structure, rook placement (open files, 7th rank, pressure on the king's flank), mobility, king safety, pawn activity, center control, threats and special endgame adjustments (bishop pair, pawn imbalances).
- Optional evaluation via a neural network (NNUE, .nnue format) through the nnue-probe library and the UCI EvalFile option, with automatic fallback to PeSTO if the network is not loaded, supporting the `seteval pesto` and `seteval material` options for debugging.
- Move generation via the standalone castro library and built-in Syzygy tablebase support.
- Tools: C, Makefile · License: MIT

gbemu

KDesp73/gbemu

- Game Boy / Game Boy Color emulator written in C and SDL3, designed with a frontend-agnostic architecture: split into a core library (`libgbemu.a/libgbemu.so`) and a command-line application (`gbemu-cli`), with support for Release and Debug builds (with sanitizers).
- Integrated graphical debugger in a standalone SDL3 window via Dear ImGui (`cimgui`): debugging state machine (pause, step, step-over, breakpoints), register/flag views, disassembler with PC highlighting and hexadecimal memory viewer.
- Full SM83 processor emulation: 256 base instructions + 256 CB-prefixed, interrupt handling (VBlank, Timer, Serial, LCD STAT, Joypad) and accurate reproduction of the HALT bug.
- Graphics processing unit (PPU) with background, window and sprite rendering, accurate scanline state transitions (OAM, Transfer, HBlank, VBlank), DMG palettes and support for the LCDC, STAT, SCY/SCX, LY/LYC, BGP, OBP0/OBP1, WY/WX registers.
- Memory Bank Controller (MBC) support: ROM-only, MBC1, MBC2, MBC3 (with RTC) and MBC5 with correct bank switching — verified via the mooneye test suite (MBC1: 11/11, MBC2: 7/7, MBC5: 9/9).
- Peripherals: timer with falling-edge counter (DIV, TIMA, TMA, TAC), joypad with row selection and active-low state, as well as serial port emulation for test ROM output.
- Save file (.sav) support with RTC data retention for MBC2/MBC3 and full machine save states (.state) (CPU, bus, timer, PPU, APU) with version control and ROM hash (F5/F9).
- Integrity checks: 85% pass rate on mooneye-test-suite (44/52) and 100% on Blargg cpu_instrs (10/10), with an automated test script (`scripts/test`).
- Tools: C, SDL3, Dear ImGui (`cimgui`) · License: MIT

scout

KDesp73/scout

- Fast, local-first search engine written in Zig, supporting local crawling, indexing and offline full-text search via SQLite FTS5.
- Indexing of titles, meta descriptions and full page content, with a configurable crawler supporting a seed URL (`--seed`), depth control (`--depth`) and continuous operation (`--infinite`).
- CLI-first design with the commands `init`, `crawl`, `query`, `list` (page/queue management), `parse` and `serve`.
- Built-in HTTP server providing a web interface for the search engine (`serve --ip --port`).
- Tools: Zig, SQLite (FTS5) · License: MIT

Gardener

KDesp73/gardener

- Embedded automatic watering and environmental monitoring system based on ESP32 (C++/Arduino framework), supporting multiple zones/plants per device and interconnecting multiple devices on the same network.
- Masterless architecture: each device operates autonomously with a unique Device ID (derived from its MAC address) and publishes data to an MQTT topic tree (`gardener/{device_id}/zone/{zone_id}/{subject}`).
- Independent soil moisture sensor (ADC), relay and pump per zone, with configurable dryness/humidity thresholds, daily schedule, maximum operating limit (safety cutoff) and minimum interval between cycles.

- Activation logic with a dual precondition (schedule + humidity threshold) and automatic deactivation upon reaching the limit or completing the maximum runtime, with manual override capability via MQTT (`cmd/water`).
- MQTT integration: publishing status, environmental measurements (DHT22) and soil moisture levels, receiving commands/settings, as well as auto-discovery (`announce`) for zone mapping.
- Parameter storage in non-volatile NVS memory (Preferences): configuration of WiFi, MQTT, Device ID and zone limits, with remote reading and updating via MQTT.
- Tools: C++ (Arduino), ESP32, MQTT, PlatformIO, Adafruit DHT, PubSubClient, ArduinoJson

artc

KDesp73/artc

- Command-line tool (CLI) for producing dynamic, animated abstract artwork via Lua scripts, with frame-by-frame rendering and export to video (`.mp4`) or looping GIF.
- Scripting API with the hooks `setup()` and `update(dt)` for full control of motion, color palettes, geometric shapes and timing.
- Rendering protocol with configurable window, background and FPS settings, as well as dynamic object management via the companion `artc.lua` library.
- Deterministic graphics generation with random seed support for exact reproduction of results.
- Export to MP4/GIF via FFmpeg integration (`-x, -F gif`) and system installation via `make install`.
- Companion tools: Artc Editor (web-based IDE for writing scripts) and `artc-service` for remote rendering execution.
- Tools: C, Lua, SDL2, FFmpeg · License: MIT

castor

KDesp73/castor

- Lightweight 3D graphics engine (raycasting engine) written in C and SDL2, designed for a retro aesthetic (Wolfenstein 3D style).
- Projection of 2D maps in a pseudo-3D environment via raycasting techniques, ensuring high FPS even on low-spec systems.
- Rendering support for sprites, floor and ceiling casting, with a modular architecture for easy extension with new game mechanics.
- Built-in unit testing system (`make -C engine test`).
- Tools: C, SDL2, SDL2_image, SDL2_ttf, SDL2_mixer · License: MIT

DataBridge

KDesp73/DataBridge

- Java library for simplifying database management: connection handling, dynamic SQL query construction, transaction support and built-in logging.
- Unified API supporting multiple RDBMS: Microsoft Access, SQLite, PostgreSQL and Oracle.
- Published on Maven Central (`io.github.kdesp73:DataBridge`) for direct integration via Maven or Gradle.
- Full documentation with online documentation and usage examples (`examples/`)
- Tools: Java, Maven · License: MIT

webc

KDesp73/webc

- C library for programmatic website construction (HTML generation) directly from C code.
- Ready-made templates for rapid development: Single Page Portfolio (`spp.h`) and Project Showcase Site (`pss.h`).
- Integration of the DaisyUI UI framework with ready-made components.
- Code documentation in header files and automatic API documentation generation via Doxygen.
- Installation system via Makefile producing static and shared libraries (`.a, .so`).
- Tools: C, CSS · License: MIT

nn

KDesp73/nn

- Neural network library written in C (Neuron/Layer/Network structures, network/data serialization), accompanied by `nni`, an interactive CLI REPL for training and evaluation.
- Support for the Sigmoid, ReLU and Tanh activation functions, with training via backpropagation and a target loss definition (`targetLoss`).
- Configurable learning rate decay, gradient clipping (`maxGradient, theta`) and bias initialization, all adjustable via a configuration file (`nn.conf`) without requiring reactivation or recompilation.
- Use of dynamic arrays (`Sizes, Doubles, Doubles2D`) for managing the architecture and input data.
- The `nni` tool provides commands for creating networks from the command line (e.g. `nn 2 8 1`), loading data, training, running predictions (`feed`) and printing status.

- Compilation into static and shared libraries (`libnn.a`, `libnn.so`) with system installation via Makefile.
- Tools: C · License: MIT

Awards and Distinctions

3rd Place — Netcompany Hackathon (Thessaloniki)

dmg-techlabs.github.io/eudi-login

- Development of *EUDI Login*, an authentication service and corresponding JavaScript SDK that leverages the European Digital Identity (EUDI Wallet) for secure authentication based on the data minimization principle.
- Implementation of an authorization flow via QR code: session initialization on the backend, scan with the EUDI Wallet, polling with a timeout (2 minutes) and automatic redirect after confirmation.
- Support for multiple digital attestations (PID, AgeOver18, IBAN, HealthID, mDL, PhotoId, TaxNumber) with configurable visibility levels (PUBLIC, ANONYMOUS_OPT, ANONYMOUS) for privacy protection.
- Security architecture using a reverse proxy: CORS control, protection against SSRF attacks (blocking internal IPs), input/output sanitization and containerization via Docker.
- Team: DMG-TechLabs (3 members) · Tools: JavaScript, PHP, Docker, HTML/CSS

Thessaloniki, October 9, 2026

The candidate
Konstantinos Despoinidis