Proposal: Maths + Programming = ❤

Personal Details

- **Full Name:** Dmitrii Kosintsev
- **Email:** dmitry.kosintsev@gmail.com
- **GitHub Username:** dmitrykosintsev
- **Your first language:** Russian
- **Location and Timezone:** Mongolia, Ulaanbaatar, GMT+8

- **Work background:** full-time mathematics teacher for Key Stage 3 and iGCSE in an international school. I currently teach 4 grade levels (6, 7, 8, and 9), with overall of 95 students.
- **Education background:** I am about to finish my BS in Computer Science. Among the courses I took at the university, there were the following electives and core courses: MATH 1281: Statistical Inference, CS 3304 Analysis of Algorithms, CS 3308: Information Retrieval, CS 3440 Big Data, CS 4407 Data Mining and Machine Learning (core course), CS 4408: Artificial Intelligence.

  Languages used during the coursework:
  Python, Java, Swift, JavaScript, Kotlin, R.

  Relevant projects:
  - a local indexer;
  - a local search system;
  - a simple vacuum cleaner AI-agent (on GitHub);

- an AI-agent using MDP to move around a given world and avoid obstacles (on
GitHub);
- UX Design Professional Certificate from Google via Coursera (design project for web and mobile
here);
- no-code pet project for my class to gamify their experience:
https://echonet.glide.page/ (based on ideas from The Gamification Guy and Gamify Glide series). The app contains bare minimum because phone usage is restricted at school;
- three.js for animation and event handling (on
GitHub);
- no experience with open source projects so far.

Project Details

Project I am interested in: **Math Games**

Motivation

As a middle-school mathematics teacher, I always look for new way to engage my students, make their learning fun and interactive, and give them enough space to practice. I found that even simple gamification works incredibly well for many students. They might not like maths, but they become eager to do anything for a reward.

As a computer-science student, I am interested in ways to automate routine processes and unleash people’s creativity. Games are perfect solution because they can implement assessments while providing tons of fun to children.

How will it impact Sugar Labs?

I like to share my knowledge and experience, and here are a few things I can bring to the organisation:

- **Network:** I teach at school with a relatively big mathematics department (5 full-time and 1 part-time teacher). This resource provides several benefits for a project like this:
  - Easy to get feedback from professionals.
  - Easy to get ideas how the project can be improved.

    - Easy to test as the games can be used in real classroom to see how children interact with them (week 11-12 is the back-to-school time).

- **Diverse experience:** I have been studying and practising programming for 3 years; however, I have got a chance to try various areas of computer science, including UX/UI design (Google UX Design Professional Certificate), ML/AI (coursework), mobile development (coursework + 100 days of Swift UI). Meanwhile, I have been teaching maths and English for the last 4 years and worked with young people for 7 years prior to that.
- **Passion:** I introduced gamification to one of my classes last November. Each student got a character with DnD-like characteristics and abilities. The last five months have shown how dedicated and hardworking engaged students can be. Now I want to share this passion with others.

What technologies (programming languages, etc.) will you be using?

Python: this language is popular and has many libraries easily available for development. Experience: **moderate** (coursework).

PyGame: Python modules that make it easy to create games.

Alternatively, I can use JavaScript + three.js to create/port my games to web. My experience with three.js includes creation of compound shapes, event handling, animations, and writing a graph visualiser.

Timeline:

Note: as a teacher, I have summer break, which is expected to last from the 9th of June until the 19th of August. This is why first and last weeks have lower time commitment.

Week 1-2: Project Planning and Setup (15-20 h)

- Define project scope, objectives, and success criteria.
- Research each math game and identify requirements.
- Allocate resources and set up project management tools.
- Finalise the technologies and frameworks to be used.

Week 3-4: Game Design and Prototyping (20-25 h)

- Design the game mechanics and user interface for each selected game.
- Create wireframes and prototypes to visualize the gameplay.
- Decide on the visual style, graphics, and audio elements for the games.

Week 5-6: Lewis Carroll's Game of Logic and Pascal's Triangle (25-30 h)

- Develop Lewis Carroll's Game of Logic.
    - Implement the game logic and rules.
    - Design the user interface and graphics.
- Develop Pascal's Triangle.
    - Implement the algorithm for generating Pascal's Triangle.
    - Create interactive visualization for exploring the triangle.
- Prepare the mid-project presentation.

Week 7-8: Nim and The Candy Game (25-30 h)

- Develop Nim.
    - Implement the game logic for Nim.
    - Design the user interface and gameplay mechanics.
- Develop The Candy Game.
    - Define the game rules and mechanics.
    - Design the user interface and visual elements.

Week 9-10: Number Guessing Game and Latin Squares (20-25 h)

- Develop Number Guessing Game.
    - Implement the logic for generating random numbers and handling user guesses.
    - Design the user interface and feedback system.
- Develop Latin Squares.
    - Implement the algorithm for generating Latin Squares.

    - Design the user interface and interactive elements.

Week 11-12: Testing, Polishing, and Deployment (15-20 h)

- Test each game for bugs, usability, and performance.
- Gather feedback from testers and make necessary adjustments.
- Polish the graphics, animations, and user experience.
- Deploy the games to an online platform or web server.
- Create documentation and instructions for playing each game.
- Conduct final reviews and ensure all deliverables are met