Pathway for Ahnik

1 Personalized Learning Approach

This course is designed specifically for Ahnik so that he develops **creativity** and **problem solving** through the lens of game development, setting Ahnik for success whatever his interests may be, coding related or not.

We will:

- Start from his current interests and past classes to expand his skills
- Use project-based learning with immediate visual feedback
- Eliminate theoretical lectures - everything will be hands-on
- Develop troubleshooting skills through guided problem-solving
- Build confidence through achievable milestones

2 Course Structure & Methodology

Learning Pathway

Weekly Sessions:

- 60 minute hands-on coding sessions
- Flexible scheduling (recommended weekly)
- Focused on student's questions and project progress

Homework Philosophy:

- Purpose-driven mini-games (not exercises)
- Designed for 15-30 minute daily practice
- Optional challenges for deeper exploration
- Midweek progress check-ins available

Learning Foundation:

- Concepts introduced through game mechanics
- Math taught visually through game physics
- Just-in-time learning - no boring lectures!

3 How to Move

- **Position:** X/Y coordinates on screen
- **Velocity:** Movement per frame
- **Control:** Arrow keys, mouse, touch
- **Project:** Move a character smoothly

4 Events & Input

- **Triggers:** Key presses, mouse clicks, collisions
- **Handlers:** What happens when events occur
- **Project:** Character jumps when spacebar pressed

5 Appearance & Animation

- **Sprites:** Loading and displaying images
- **Frames:** Simple animation cycles
- **Project:** Walking animation when moving

6 Game Logic

- **Variables:** Score, health, timers
- **Conditionals:** If-then rules
- **Project:** Collect coins to increase score

7 Interactions

- **Collision:** Rectangle and circle detection
- **Triggers:** Zone-based interactions
- **Project:** Enemy disappears when jumped on

8 Game States

- **Screens:** Menu, Play, Game Over
- **Transitions:** Changing between states
- **Project:** Restart game after Game Over

9 Final Project: Build a Game

Create a complete game with:

- **Complexity:** At least Ping Pong level
- **Required:**
  - Player control
  - Scoring system
  - Win/lose conditions
  - Visual feedback
- **Examples:**
  - Pong with power-ups
  - Platformer with 3 levels
  - Space shooter with enemy waves

10 Transition to Python

- **Why Python?**
  - **Saving:** Store game data permanently
  - **Multiplayer:** Networked games
  - **AI:** Smart enemies/opponents
  - **Performance:** Handle complex games
- **First Python Project:** Rebuild your Scratch game in Python

11 Next Steps

- **Python Track:** Continue with Python game development
- **Advanced Topics:**
  - Physics engines
  - Online multiplayer
  - 3D graphics
  - AI opponents

Disclaimer: After completing this core curriculum, you'll transition to the Python track where we'll recreate your Scratch projects with real code and add advanced features.