The Numbers Inside Video Games

(Also available in WeScheme)

Students play a simple video game, identifying which components are constant, which are variables, and how they change. They discuss their favorite games and think about the work involved in making them.

Lesson Goals	Students will be able to: • Identify the objects in a video game that are changing. • Use math language to describe what is changing about each object. • Understand the time, money, and resources it takes to create a popular video game.
Student-Facing Lesson Goals	• Let’s identify the objects in a video game. • Let’s use math vocabulary to describe what is changing about each game object. • Let’s think about the time, money, and resources it takes to create a popular video game.
Materials	• PDF of all Handouts and Page • Ninja Cat Game • Lesson Slides • Printable Lesson Plan (a PDF of this web page)
Key Points for the Facilitator	• Students will need their own Google accounts. • Take care to manage student expectations about what their game will be like. Modern games are very complex!

Glossary

coordinate :: a number describing an object’s location

variable :: a name or symbol that stands for some value or expression, often a value or expression that changes

Reverse Engineering a video game *25 minutes*

Overview

Students play a simple video game, and gradually break it down into parts. Doing so reveals how ***coordinates*** play a crucial role in video games, and how animation is created via equations that govern the changing values of those coordinates.

Launch

Play the Ninja Cat Game onscreen while students watch. Purposely make mistakes while playing the game, which should elicit responses/direction from students.

- Take turns playing the game in pairs.
- After you've both had a chance to play, write down what you *notice* about the game on Notice and Wonder.
- "Notice"s should be statements, not questions. What stood out to you? What do you remember?

Crowdsource students' Notices.

What do you *wonder* about the game? What questions do you have about how it works? Write these down on Notice and Wonder.

Crowdsource students' Wonders.

Pedagogy Note: Notice & Wonder

This pedagogy is a widely-used best practice in Math-Ed, and is used throughout this course. In the "Notice" phase, students are asked to crowd-source their observations. No observation is too small or too silly! By listening to other students' observations, students may find themselves taking a closer look at the game. The "Wonder" phase involves students raising questions, but they must also explain the context for those questions. Sharon Hessney (moderator for the NYTimes excellent What's Going On in This Graph? activity) sometimes calls this "what do you wonder…and **why**?". Both of these phases should be done in groups or as a whole class, with adequate time given to each.

Investigate

Work with your partner to complete Reverse Engineer a video game.

- 1st Column: What are all the various *things* in this game? *(A dog, Clouds, etc.)*
- 2nd Column: For each of those "things", what is changing about them? *(Location, Value, etc.)*
- 3rd Column: For each change, how is it modeled mathematically? What ***variable***(s) does the program need to keep track of? *(x-coordinate, y-coordinate, amount, etc.)*

Common Misconceptions

- Students are likely to describe what the character is *doing*, as opposed to *what changes*. For example: "The dog is moving to the left" is not actualy describing the property being changed (position, place, location, etc).
- Students may write down what they *hope* is changeable, as opposed to what actually changes. It's common for students to say the cat's costume is changing, because they assume the cat will somehow "level up" if they get enough points.

Synthesize

The main idea here is to understand that while we see images on a screen, the computer only sees a small set of variables, which uniquely model the state of the game. The way those numbers change determines how the game behaves, and we can add features to the game if we're willing to keep track of more variables.

- If each character is represented by both an x-coordinate and a y-coordinate (some of which are fixed and some of which are randomly generated), how many numbers does it take to represent a single frame of the video game?
  - *Nine: 2 for the cat (moves left and right), 2 each for the cloud and ruby (move left at random y-positions), 2 for the dog (moves left at a fixed y-position) and 1 for the score*
- How are those numbers changing - or *varying* - as the game plays? When do they increase? Decrease?
  - *The cat's y-coordinate increases when they jump and decreases when they fall. x-coordinates increase when a character moves right, and decrease when they move left. The score increases by 1 when the cat gets a ruby and by 10 when the cat jumps on the dog.*
- How many variables would we need to have a two-player game with two cats?
  - *(Sample answer) Eleven: we'd need to add another set of x-y coordinates.*

- How many variables would we need if the entire game was in 3d?
    - *(Sample answer) Fifteen: we'd need to add z-coordinates to each character.*
- How many variables would we need to keep track of in a modern video game?
    - *A whole lot — Thousands. Millions!*

Connecting to Real Games *25 minutes*

Overview

Students apply this way of thinking to more complex, real-world games. They also get a sense for how much work is involved in creating games like that.

Launch

Ask students to share out their favorite current video game. Write the names of the games on the board.

Investigate

Let’s choose one current, popular game to discuss.

- How long do you think it took to create that game?
- How *many people* do you think it takes to create a game like that?
- How *much money* does it take to create a game like that?

Collect students estimates for each of the questions above.

Optional: Once students have made their estimates, have students use the Internet to research these questions and compare the actual numbers to their estimates.

Answers: There’s a lot of variability, especially between game consoles and cell phone games! But here’s a few sample numbers:

Title	Time	Team Size	Budget
Call of Duty: Modern Warfare 2	2 years	500+	50m+
Final Fantasy VII	3 years	100+	40-45m

Title	Time	Team Size	Budget
Shadow of the Tomb Raider	3+ years	100+	75m+

The goal here is not to discourage students from the possibility of eventually creating a game like their favorite game, but to manage expectations given our limited resources (time, money, and people). By starting with this game project, students are learning transferable skills that can help them later on in learning new programming languages and creating bigger projects.

Synthesize

- Share-back: have students share their estimates with the class. Was anything drastically higher or lower than they expected?
- What does this tell us about making modern games?
- Are we likely to create games like the ones you researched?

The 3d, two-player version of NinjaCat needed a lot more numbers than the simple one you saw here, *but the actual concepts at work are the same*. Even if we don't have time to make games like the ones we chose here, you'll learn the same concepts just by making a simpler one.