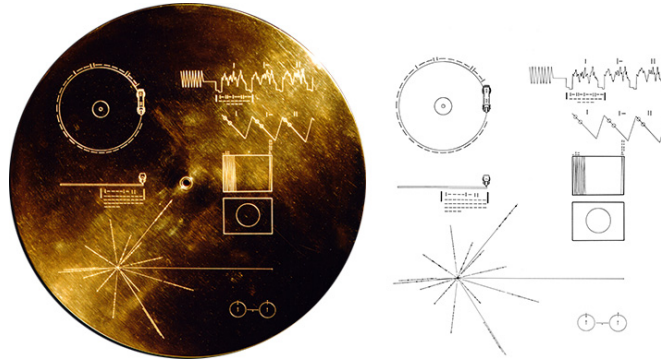




Telling the Universe Who We Are

4th/5th Grade - Computational Thinking Unit Add-On

2023-2024

Unit Overview:

The Voyager space probe was launched by the nations of the world, to go beyond our solar system and into the universe. It was decided that on board, it would carry a circular disc, made of gold, with information about the people of Earth, and even encoded sound of many languages being spoken. We don't know who may find this message some day, but we wanted them to know something important: who we are.

Now we have the ability to send messages directly into space. It may take millions of years, but these messages will eventually reach the many planets in the universe with climates and atmospheres like Earth's. What would you like to tell the universe about who you are? About what is important to you? About what you value, what you believe, and what you enjoy? If someone, some day, sees your message, what can you show them or tell them so that they know you?

In this unit, students will build up familiarity with the foundational elements of Scratch programming (events, control statements, loops, coordinates, broadcasting, resetting, parallelism, and sprite object orientation; see glossary below) to prepare them for the final project: code a project to explain to the universe who you are. The unit culminates with a final student showcase where students will present and share what they have built in Scratch to bring aboard the Voyager.

Unit at a Glance

Unit Question: How can we use computer programming to express who we are?


 Assessment


 Hands On Prep Required


 Computer Use

Big Question #1: How can we create and share projects about ourselves that we interact with by clicking and typing?

Progress Build 1: Students will be able to create code sequences, using Scratch’s Event blocks and other blocks, to make sprites respond to user interactions in order to communicate their name to others.

0	1	2	3
Warm up: Block-Based Coding 	10 Blocks  	Animate My Name: Events  	Animate My Name: Sharing 

Big Question #2: How can we code the position and timing of events to share information about our lives?

Progress Build 2: Programmers can collaborate to create coding projects about a virtual pet and scene-based storytelling, using a variety of Scratch block types to coordinate the position and timing of sprite actions.

4	5	6	7
Virtual Creature Design 	Virtual Creature Broadcasts and Variables   	Storytelling with Timing 	Storytelling with Multiple Scenes  

Big Question #3/Final Challenge: How can we use technology to create projects that express who we are?

Progress Build 3: Programmers use imagery, text, and a variety of Scratch block types to create projects that capture aspects of who they are.

8	9	10	11	12
About Me: Idea Generation 	About Me: Building 	About Me: Building 	About Me: Building 	About Me: Showcase  

Standards Overview

<u>CSTA Standards Alignment</u>	<u>ISTE</u>
1B-AP-10 Create programs that include sequences, events, loops, and conditionals. 1B-AP-11: Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process. 1B-AP-15 Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended. 1B-AP-16 Take on varying roles, with teacher guidance, when collaborating with peers during the design, implementation, and review stages of program development. 1B-AP-17 Describe choices made during program development using code comments, presentations, and demonstrations. 1B-IC-21 Use public domain or creative commons media, and refrain from copying or using material created by others without permission. 1B-NI-05 Discuss real-world cybersecurity problems and how personal information can be protected.	1.1c Feedback to Improve Practice: Students use technology to seek feedback that informs and improves their practice and to demonstrate their learning in a variety of ways. 1.2.c Intellectual Property: Students demonstrate an understanding of and respect for the rights and obligations of using and sharing intellectual property. 1.2.d Digital Privacy: Students manage their personal data to maintain digital privacy and security, and are aware of data-collection technology used to track their navigation online. 1.3.c Curate Information: Students curate information from digital resources using a variety of tools and methods to create collections of artifacts that demonstrate meaningful connections or conclusions. 1.4.c Prototypes: Students develop, test and refine prototypes as part of a cyclical design process. 1.5.d Algorithmic Thinking: Students understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions. 1.6.b Original and Remixed Works: Students create original works or responsibly repurpose or remix digital resources into new creations. 1.7.c Project Teams: Students contribute constructively to project teams, assuming various roles and responsibilities to work effectively toward a common goal.
<u>NGSS</u>	Computational Thinking
3-5-ETS1-1. Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost. 3-5-ETS1-2: Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem. 3-5-ETS1-3: Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.	CT CONCEPTS/SKILLS: Algorithms - making steps and rules Abstraction - removing unnecessary detail Logic - predicting and analyzing Patterns - spotting and using similarities Evaluation - making judgements Decomposition - breaking down into parts CT VALUES: Persevering - keeping going Tinkering - changing things to see what happens Collaboration - working together Creating - designing and making Debugging - finding and fixing errors

Assessments

Lesson	Assessment:	Grading Support
1	10 Blocks project (formative) ☐ Uses at least 5 of the 10 foundational blocks	Link to rubric and exemplar
2	Animate My Name project (formative) ☐ 2+ sprites that appear as letters (can be first name, nickname, initials, or something else) ☐ Each letter sprite has at least one event block, which is connected to a sequence of coding blocks ☐ Each letter sprite can be interacted with in some way (by clicking it, by typing a key, by clicking the green flag, by another sprite touching it, or something else) and has a visual or audio response	Link to rubric and exemplar
5	Virtual Creature project (formative) ☐ 1 sprite representing an animal or character that is the “virtual creature” ☐ 2-6 other sprites that represent items the virtual pet can interact with ☐ Each item sprite can be interacted with in some way ☐ Clicking the green flag resets the virtual pet and all other sprites to their starting position and appearance	Link to rubric and exemplar
7	Scenes project (formative) ☐ Uses 1+ “Wait” blocks to coordinate the timing of sprite actions ☐ Uses 1+ “Broadcast” blocks to coordinate the timing of sprite actions ☐ Uses 2+ backdrops on the stage to represent scenes ☐ Clicking the green flag resets all sprites to their starting position and appearance	Link to rubric and exemplar
12	About Me project (summative) ☐ Communicates what is important to you through all of these: ☐ images ☐ text (appearing in sprites, or using Say/Think blocks) to communicate to the user ☐ actions and events ☐ Uses event blocks to control sprite action; has at least two ways that users can cause events in the project to occur ☐ Uses control blocks (wait, broadcast, repeat, forever, if/then) to coordinate the action of different sprites ☐ Uses motion blocks to animate sprites ☐ Uses resetting to ensure the project runs the same way if it is run two times in a row ☐ Has clear instructions for using the project	Link to rubric and exemplar

Unit Vocabulary

(vocab cards for word wall available here)

Session #	Vocabulary; Purple = Scratch-specific meaning
0	code: instructions used to control a computer (synonym: software) programming: writing and changing code to control a computer programmer: anyone who creates or modifies a computer program (synonym: software engineer) code blocks: code shapes that you snap together, rather than type out blocks library: the area in Scratch where you get code blocks to use (on the left side) coding area: the area in Scratch where you assemble code blocks together (in the center) stage: the area in Scratch where a project runs (on the right side)
1	computational thinking: the process of describing and solving problems by breaking them down into simple steps. It is a powerful problem-solving technique that equips us to solve complex problems in the modern world algorithm: seeing what's the same and what's different to anticipate, recognize, and predict; using prior experiences to help solve a problem abstraction: the process of filtering out – ignoring - things we don't need in order to concentrate on those that we do. sequence: an ordered series of steps or actions Motion blocks: Scratch blocks controlling sprite movement (blue) Looks blocks: Scratch blocks that change sprite appearance (purple) Sound blocks: Scratch blocks related to audio (pink/magenta) Event blocks: Scratch blocks that trigger code to begin running (light yellow/orange) Control blocks: Scratch blocks managing how code runs (dark yellow/orange)
2	sprite: object in a program that has its own code, and its own appearance debug: figure out why code isn't doing what you want, and fix it
3	pattern recognition: seeing what's the same and what's different to anticipate, recognize, and predict; using prior experiences to help solve a problem private: only the project creator can view and run it public: anyone can view and run the project sharing: allowing public access to a project studio: a collection of public Scratch projects remix: making a copy of someone else's Scratch project, and modifying it
4	tutorials: guides and videos demonstrating what you can build with Scratch loop: code that runs multiple times
5	broadcast: when an event is sent in Scratch to all parts of the project variable: a storage container that holds a value that can be modified or retrieved by your program
10	resetting: steps taken to make sure a project runs the same way every time