BUILDING SYSTEMS FROM SCRATCH CURRICULUM GUIDE

TERC Because math and science build futures

Northeastern University

BUILDING SYSTEMS FROM SCRATCH

CURRICULUM GUIDE

TABLE OF CONTENTS

Introduction ____ 6

MODULE ONE ____ 15

Section 1: Introducing systems ____ 15

Section 2: Climate change as a system ____ 17

Section 3: More about climate change ____ 20

MODULE TWO ____ 24

Section 4: What are games? ____ 24

Section 5: Designing in Scratch ____ 28

MODULE THREE ____ 36

Section 6: What's my game? ____ 36

Section 7: Game design to learn the details of climate change ____ 40

Section 8: Completing and showcasing games ____ 47

Handouts and Worksheets ____ 49

The curriculum was funded under NSF grant #1542954. The design decisions are those of the authors and do not necessarily reflect those of the National Science Foundation.

Introduction

The Building Systems from Scratch curriculum is designed to support integrated learning by interweaving game design, climate science and computational thinking practices. Students design and create games in Scratch, a block-based programming language, to teach others about climate change. Students explore a systems approach to learning about climate dynamics and climate change that integrates humans as a central component.

The project believes that when young people build games, they construct knowledge at the same time. The instructional approach adopts a view of classrooms as networks of expertise distributed among students.

Three modules engage students with jigsaw learning about climate science, understanding the basics of systems thinking, principles of design for serious games, and learning computational thinking practices through programming in Scratch.

Instruction includes:

- Student-led research of climate science based on videos, simulations, and modeling activities
- Collaborative pair programming for game design
- A focus on game design consisting of three related elements: Reality (climate content), Meaning (what the player is intended to learn), and Play (how the player will engage with the game).

Length of unit

We estimate that the unit will take about 23 55-minute class periods, or more depending on how much time students will need to design and code their games.

Module	Topic	Number of class periods
One	Introducing systems; climate as a system; more about climate change	6
Two	What are games; designing in Scratch; what's my game	4
Three	Game design; showcasing games	10-12

Rationale for the design of the curriculum

Why games – and why Scratch

Game design is highly engaging at middle and high school levels — as a way of using computational languages, of playing with and representing complex ideas and building aspects of students' own identities. Research shows that students also learn STEM content from designing games.

Scratch provides programming blocks that "snap together" and communicate what should be represented on the screen. The blocks support thinking systematically, or computationally, by creating specific logical sequences. Scratch supports playful iteration with designs, and does not require formal programming skills.

Triadic game design (Harteveld, 2011) requires that student designers balance reality, meaning, and play for the successful creation of educational games. *Reality* relates to the domain–climate change, *Meaning* relates to the ideas that players are meant to get from the game, and *Play* relates to effective game design-how the player experiences gameplay.

Computational thinking

Building systems tries to address this problem by integrating computing in science, in which students use technology in a meaningful context—education about climate change—that is engaging—designing a game, and serves a useful goal—teaching others about the topic.

Our research has shown that this type of design supports student learning of computational thinking in addition to learning about climate change. As they represent a climate system or subsystem, students break down the system into parts and interactions ("abstraction and decomposition"), problem-solve as they plan to represent these in their game ("logical thinking and problem-solving"), and code to create the game ("algorithmic thinking").

What is systems thinking

Systems thinking is fundamental to learning about climate change and other complex topics. Systems thinking suggests that studying interrelationships among the parts of a system rather than isolating the parts and looking at them separately is the best way to understand a complex system.

Some key concepts of systems thinking in the curriculum are:
All systems are composed of interrelated components. A change to any component or any connection between two components affects the entire system. An excellent example of an early surprise that the systems approach revealed was that the actual temperature increase on Earth is greater than expected if only carbon dioxide levels were considered. What is a surprising greenhouse gas? Water vapor.

Interrelationships determine behavior. The behavior of a system depends on interrelationships—or causal connections— among components because these connections usually determine how the system "behaves."

System behavior is an "emergent phenomenon." How a system behaves can't simply be determined by looking at its components alone. The behavior of a system only emerges when the parts interact as a wider whole.

As designers create games, and see what happens when adding or removing game components, they learn about a specific system from *reality*—like climate. Climate systems may be represented any number of ways within a game, depending upon the designer's understanding, choices and preferences. Games are often *models* of something in the world—and are usually interactive models. Also, games are essentially systems themselves.

What are participatory pedagogy and distributed expertise

Research has shown that using participatory pedagogy for game design allows students to shape content, collaboratively solve problems, and develop confidence and aptitude in making games. Participatory pedagogies encourage reliance on distributed expertise because responsibility for learning resides with all participants. Knowledge is spread across teacher, students, groups. This approach allows students to take ownership of their work and contribute to a shared purpose.

However, participatory pedagogies require teachers to change their roles and relationships to students and to knowledge. Participating in school in this new way is unfamiliar to students too. It means that students need to be comfortable with making mistakes, trying things out, struggling, and learning from their peers.

Our work has shown that teachers with little experience computing can successfully implement the programming and game design parts of the curriculum by leveraging distributed systems of expertise among students through online search strategies, curricular resources, and identifying students as experts publicly to their peers.

Learning outcomes

The prerequisites for student understanding to be able to engage with this unit are:

- The difference between weather and climate
- The factors that determine climate

The overarching learning outcomes for the unit listed below fall into four main areas: i) climate change systems, ii) Computational thinking, and iii) Game design.

i) Climate change systems:
The climate change learning outcomes shown in the two tables below are differentiated into overall skills related to systems thinking, and climate change concepts.

Systems skills
Students will be able to: • Identify stocks and flows in the climate system • Represent the causal relationships that exist among the parts • Show how causal relationships among the parts generate dynamic system behavior (e.g., reinforcing and balancing feedbacks) • Create a systems diagram • Describe what might happen if a component of the climate system changes.

Details of the carbon cycle (also a system)
○ Important components of the **carbon cycle** that act as stores or "reservoirs" of carbon are Biomass, oceans, fossil fuels, the atmosphere, soils. ○ More CO_2 is released in the atmosphere **today than in the past** because of burning of fossil fuels. Too much CO_2 in the atmosphere is responsible for the greenhouse effect (along

with other ghgs).

Earth's energy budget, and how atmosphere, ocean (and cryosphere), pedosphere, and biosphere interact (the global system)

- "The Greenhouse Effect" refers to the **insulating effect** of ghgs - water vapor, CO_2, methane – in our atmosphere.
- Some of the **solar radiation** reaching Earth's surface is absorbed. It turns into thermal (or heat) energy. The rest is reflected or re-radiated back into the air as infrared radiation (IR).
- The amount of IR that is re-radiated by a surface depends on its "**albedo.**" Ice reflects high levels of IR, it has a high albedo. Water absorbs more IR and a lower albedo.
- The majority of the thermal energy at the Earth's surface is stored in the ocean.

Impacts of climate change

- **Biotic impacts** include shifts of the ranges in which plants and animals live, changes in seasonal behavior (e.g., flowering earlier in spring, birds migrating earlier) and extinction.
- **Abiotic effects** include melting land ice causing rising sea levels, loss of sea ice allowing more heat to be absorbed by polar oceans, shifts in precipitation patterns, increased intensity of heat waves, floods and storms.

Carbon reduction, mitigation and adaptation strategies

- **Carbon reduction** refers to ways of slowing or reversing global warming through removing ghgs from the atmosphere (e.g., through tree planting, capturing emissions from industry)
- **Mitigation** refers to ways of slowing or reversing global warming through reducing emissions (e.g., through energy efficiency, renewable energy, eating less meat), or reducing the amount of solar energy that gets trapped as heat (e.g., painting roofs white).
- **Adaptation** refers to ways of coping with the impacts of climate change, e.g., building sea walls, capturing rainwater run-off in dry areas.

ii) Computational thinking (CT)

The following table lists the CT practices students engage in and identifies the activities in which they occur.

CT Category	Practices	Example: Representing climate systems
Systems Thinking	Investigate system functioning	-Identify (decompose) components of system, e.g., fossil fuels, CO2 -Identify feedback relationships among components, e.g., ice-solar radiation-albedo-ocean temperature -Understand a game as a system of interacting components
Modeling	Model design and construction	-Design and construct a game as a model of systems phenomena so that the player understands the systems concept
Computational Problem Solving	Problem decomposit-ion	- Decompose problems into sub-problems -Simplify complex problems so that mapping of features onto computational solutions is more accessible
	Logical thinking	- Use if, if else, logic operations, e.g., to support feedback interaction

	Data representat-ion	- Create and operate on variables, lists to introduce system complexity
	Flow control	- Use repeat, forever commands to control sequencing, or flow
	Abstraction	-See relationship between game event and related code (supports remixing)
	Debugging	-Determine why components of game do not work
Communication		-Conduct peer critique -Describe game and system in game to an audience -Explain connection of system represented in game to the "big picture" of climate change

iii) Game design
These skills are essential for the development of good educational games.

Game design
Students will be able to: • Classify different types or genres of games, and identify their characteristics • Describe the characteristics of educational games that make them engaging and educational. • Use triadic game design – focusing on Reality, Meaning and Play – to create their games

Instructional Approach

Students can learn in many different ways

Although students will be designing games that represent only a part of the complex climate change "story" – and will therefore learn in more depth about that part - the curriculum is designed so that they learn about climate change from other activities as well. These include:

- **Learning library:** During Section 3, students take responsibility for teaching each other the basics of climate change topics as represented in the topic overviews in the Learning Library
- **System diagramming:** Each time students revisit their systems diagram they have an opportunity to incorporate more climate change content
- **Peer programming:** A structure for supporting peer programming is explicitly laid out with very clear rules about how students should contribute to their joint project (Section 5). Scroll down the page to "The basics of computer science" to find the video. Students take turns as "Driver," in charge of the keyboard, while the other student is the "Navigator," who can make suggestions, point out potential problems and keeps the big picture in mind. The video outlines rules (at the 2:39 mark) which you can post in the classroom. However, some student pairs can collaborate well by dividing their design tasks, e.g., one student does the coding, while the other student pays attention to aesthetics or the look and feel of the game. Since one of the goals is for all students to learn to code, the teacher will need to exercise judgement about how best to support student collaboration depending on the dynamics in individual pairs.

- **Learning from playing and critiquing games:** Students learn different content from playing and critiquing each other's games. Teachers should provide explicit encouragement for students to play each other's games and/or talk to each other about ideas, and build in 5-10 min a day towards the end of the unit. Teachers should also encourage self-directed spontaneous critique.
- **Supporting documents:** Design templates, critique forms, and rubrics are all carefully designed to support students to deepen their knowledge about climate change
- **Visuals:** You may find it useful to put up good visuals about various climate change concepts around the classroom. These visual cues can help with the cognitive load of drawing on relevant systems thinking and climate change concepts all while designing a game, thinking about Reality, Meaning and Play, and keeping the educational intention of the game in mind.
- **Mini-lessons:** Student pairs will be drawing on different content to design their games. If you feel that a number of students could use a little extra support – either with programming or with particular climate change content - one possible activity structure to use here is a "mini-lesson" in a corner of the room. Without distracting the rest of the class, you can bring a small group of students together to quickly problem-solve or to explore a topic in more depth, with your support as/if needed.
- **Experts:** Based on a quick survey before the unit begins (see Handouts section), identify if there are students who have had experience coding, with Scratch or other platforms. Check with them that they feel comfortable being identified to the class as an "expert." Introduce the expert users as another resource in the classroom in a way that creates a collaborative environment in which they are approachable by all students. Expert students should be coached with the following tips (all supported elsewhere in the curriculum as well):
    - They should tell the student they are helping *where to look for an answer*, e.g., the tabs in Scratch, the Scratch binder of help cards, searching online
    - They should *suggest actions that the student can try*, rather than telling them step by step while they do the steps, or putting their hands on the keyboard themselves
    - They should suggest *remixing* – finding code that already exists in a similar project in the studio or on the Scratch website.

Getting started and selecting student pairs

A few days before starting the unit, survey students in the class to determine what level of experience they have with Scratch. This provides useful information for setting up student partners. Try to pair students of differing abilities, e.g., so that at least one partner has some experience with Scratch, or an academically strong student with a student who is struggling. If none of the students have coding experience, use your usual methods for creating pairs.

The survey will also help to identify those students who can provide programming expertise to their peers. See "Experts" in preceding section.

To support student work in programming pairs, review a video made by the Scratch education community of two students who explain their preferred model for working together. It can be found at: *https://code.org/educate/resources/videos (2:50 min).*

Strategies for when students get stuck

To support students to participate in their own ways, and make their own decisions, there are several ways to help them - particularly when they get stuck with programming challenges. Here are some of the ways that we have found work well:

i. Encourage students to click around in Scratch, explore the tabs, see what happens. There is a back button if they change some code and don't like the result.

ii. Suggest that students consult the binder of Scratch tips, selected to help support the kinds of games they might want to build.

iii. Encourage them to do searches on Google – there are a lot of resources available, e.g., how-to videos, images they can import into Scratch, extra science content, etc.

iv. Remind them that they can ask the students you identified as experts in the class.

v. Suggest they visit the Scratch studio to see how other designers solved problems.

vi. Encourage them to "Remix" – using code that someone else has developed already to do something that they want to do saves time, and is an established good practice in computer programming. Scratch supports remixing through the "Backpack" function.

Formative evaluation

Opportunities for formatively evaluating how students are doing:

1. **Systems diagrams (concept maps):** Sections 1, 2, 3 and 7, students create and then revisit and elaborate a systems diagram of their current understanding of climate change. These can be used to track progress.
2. **Student presentations:** In Section 3, students prepare presentations on climate change concepts.
3. **Design template:** In Section 6, students use a design template to plan out their ideas for their game. The teacher can either collect and review these herself, or ask students to exchange them for critique.
4. **Student critique:** During game design, atudents have several opportunities to formally critique and write up the critique of each other's games. Also, the teacher can circulate and review games as they are being developed. The teacher can remind struggling students to consult their expert peers in programming, or point their attention to specific resources, e.g., the learning library, the Scratch studio and binder, google, etc. (See Strategies for when students get stuck, above.) The teacher will also find that students will start to critique each other's games spontaneously – this should be encouraged!
5. **Exit tickets:** The curriculum suggests exit tickets after each lesson and these also provide information on how students are learning and what they might need support with. Note that they are intended to be optional, to be used according to your usual practice.

Teacher support

Creative Computing Teacher Guide

Extensive resources to support the Scratch activities can be found online. A community has grown around the use of Scratch in the classroom, and resources and tips suggested by this teacher community are available at: http://scratched.gse.harvard.edu

Among the resources on this site is an extensive Creative Computing Teacher Guide, which you should download and familiarize yourself with to the extent that you can point students to particular resources if they get stuck. It can be found at: http://scratched.gse.harvard.edu/guide/

There is also a handy spreadsheet created by the folks at Scratched that can be quickly and easily scanned to locate a particular resource: https://goo.gl/EWbzVz

Standards alignment

The Learning Goals described above are related to the following standards in the Next Generation Science Standards (Achieve, 2013):

Scientific practices	**Crosscutting Concepts**
1. Asking questions (for science) 2. Developing and using models 3. Planning and carrying out investigations 4. Analyzing and interpreting data 5. Using mathematics and computational thinking 6. Constructing explanations (for science) 8. Obtaining, evaluating, and communicating information	1. Patterns. 2. Cause and effect: Mechanism and explanation. 3. Scale, proportion, and quantity. 4. Systems and system models. 5. Energy and matter: Flows, cycles, and conservation. 6. Structure and function. 7. Stability and change.
Content standards *ESS2.A: Earth's Materials and Systems* All Earth processes are the result of energy flowing and matter cycling within and among the planet's systems. This energy is derived from the sun and Earth's hot interior. The energy that flows and matter that cycles produce chemical and physical changes in Earth's materials and living organisms. (MS-ESS2-1) *ESS3.C: Human Impacts on Earth Systems* • Human activities have significantly altered the biosphere, sometimes damaging or destroying natural habitats and causing the extinction of other species. But changes to Earth's environments can have different impacts (negative and positive) for different living things. (MS-ESS3-3) • Typically as human populations and per-capita consumption of natural resources increase, so do the negative impacts on Earth unless the activities and technologies involved are engineered otherwise. (MS-ESS3-3),(MS-ESS3-4) *ESS3.D: Global Climate Change* Human activities, such as the release of greenhouse gases from burning fossil fuels, are major factors in the current rise in Earth's mean surface temperature (global warming). Reducing the level of climate change and reducing human vulnerability to whatever climate changes do occur depend on the understanding of climate science, engineering capabilities, and other kinds of knowledge, such as understanding of human behavior and on applying that knowledge wisely in decisions and activities. (MS-ESS3-5)	

And, finally, two quotes from the Next Generation Science Standards:

> "**Systems and System Models** are useful in science and engineering because the world is complex, so it is helpful to isolate a single system and construct a simplified model of it. To do this, scientists and engineers imagine an artificial boundary between the system in question and everything else. They then examine the system in detail while treating the effects of things outside the boundary as either forces acting on the system or flows of matter and energy across it—for example, the movement of carbon from live organisms

or from fossil organisms (fossil fuels) into the atmosphere." (www.achieve.org, Appendix G, p. 17).

"**Models** can be used for understanding and predicting the behavior of systems. Models can be used to represent systems and their interactions—such as inputs, processes and outputs—and energy and matter flows within systems. Systems may interact with other systems; they may have sub-systems and be a part of larger complex systems. Models are limited in that they only represent certain aspects of the system under study." (*http://ngss.nsta.org*).

The International Society for Technology in Education (ISTE) and the Computer Science Teachers Association (CSTA) collaborated to define computational thinking in an "Operational Definition of Computational Thinking for K-12 Education" (iste.org) in a way that should be recognizable to and usable by teachers. The elements of CT in their definition that are addressed by the curriculum are:

- Formulating problems in a way that enables us to use a computer (including humans) to help solve them
- Representing data through abstractions such as models and simulations
- Automating solutions through algorithmic thinking (a series of ordered steps)
- Identifying, analyzing, and implementing possible solutions with the goal of achieving the most efficient and effective combination of steps and resources
- Generalizing and transferring this problem-solving process to a wide variety of problems.

Reference

Harteveld, Casper (2011) Triadic Game Design: Balancing meaning, reality and play. New York NY: Springer.

MODULE ONE
Section 1: Introducing systems

(1-2 class periods)

Overview

In this section, students will explore how a classroom is a system, and diagram the system they have brainstormed as a concept map, which we will also call a "systems diagram."

Learning goals

- Students will be able to identify components and functions in a simple system, and create a systems diagram to represent them
- Students will be able to create a systems diagram that represents their current state of knowledge about climate change.

Do Now

Students write a brief response to the question: What do you want to learn about climate change?

Activity 1: Setting the context for the unit

Teacher sets the context to the overall task for students in this unit. Students will be creating a game in Scratch that is designed to teach other students about climate change. To do this, they will be learning about systems and climate change to incorporate into their games. Along the way, they will be learning Scratch, and learning about what makes a good game. Once they have acquired the necessary knowledge and skills, they will create their games. Scratch is a tool that both drives what students need to learn and allows them to communicate about what they have learned.

Activity 2: How is a classroom a system?

1. Since climate is essentially a very complex system that operates on a global level, the class begins by thinking about a simpler system closer to home. In small groups, students brainstorm about the question: "How is this classroom a system?"
The teacher collects their ideas on the board, and facilitates a whole-class discussion. Any responses are appropriate at this point, including both physical, e.g., "sunlight," and nonphysical, e.g., "student ideas," components.

Prompts for the task:

- What are things that come into the classroom?
- What are things that go out of the classroom?
- What are things that move around in the classroom?
- What things are needed to keep the classroom going? (e.g., keeping warm, etc.)

2. Students construct a system diagram using a concept map to represent the systems components of the classroom. Students not familiar with concept mapping will need support in listing all the components they think are relevant to a system, and in representing the connections among them. They should draw arrows to try to represent all the **causal connections** that they recognize among the components. **Note that the direction of the arrows should represent causation, e.g., X causes Y has an arrow pointing from X to Y.**

Activity 3: What do we know about climate change?

Have students divide into small groups, brainstorm, and make notes about the question:

- What is climate change, and what causes it?

As a class, students create a systems diagram to organize these thoughts.

Note: At this point students may or may not understand the concept of feedbacks.

Teacher Note

The goal of activity 3 is to establish a baseline for what knowledge students bring to the unit now. In the rest of the unit, their diagrams can serve as an ongoing formative evaluation resource for the teacher. Because students will revisit and expand on their diagrams regularly throughout the unit as their knowledge and understanding of Earth systems dynamics grows, teachers can track the development of their understanding, and use this information to decide on the additional resources to provide, if necessary. Therefore, teachers should elicit student ideas, and try not to use this as an opportunity for instruction. Students will have plenty of opportunity to develop their ideas and change the diagram as they learn more.

Exit ticket

Class period 1: Students give an example of another system and explain why the think it is a system.
Class period 2: List two things you learned about climate change that you did not know before.

Section 2: Climate change as a system

(1 class period)

Overview

In this section, students are introduced to terminology relevant to climate change that allows them to represent the system as a "stock and flow" model, and to the "bathtub model," an analogy for how the carbon cycle has become unbalanced due to excess flow of CO2 into the atmosphere. They revisit the systems diagram they created earlier to explore what happens when one component of the system is changed.

Learning goals

- Students will be able to identify the stocks and flows in their systems diagrams and describe what might happen if stocks or flows in the system are changed.
- Students will be able to explain the bathtub model of the carbon cycle.

Do Now

Students brainstorm and write a paragraph on why they think climate change can be represented as a system.

Activity 1: System terminology

Introduce students to the characteristics of complex systems. Complex systems have:

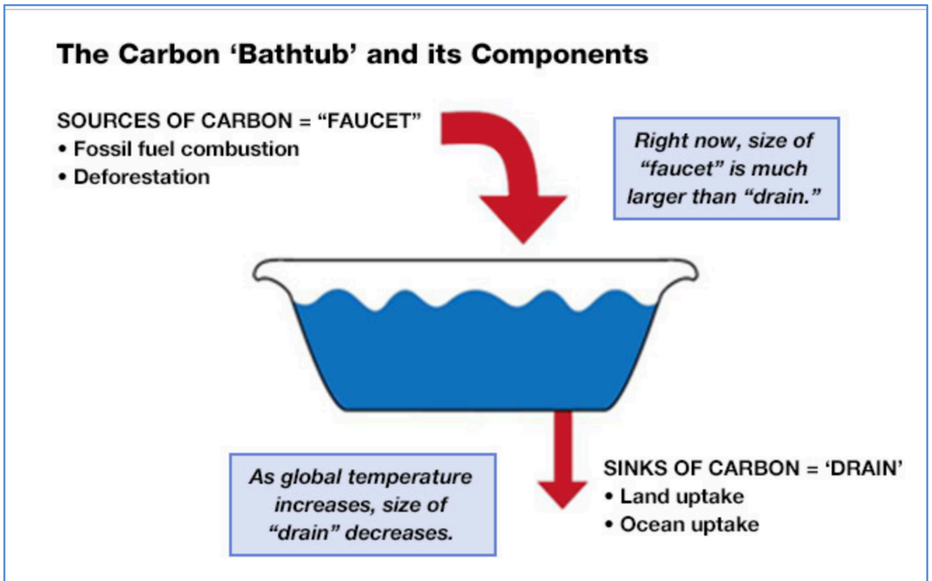
- **Components or Stocks**=A systems term that applies to a part of a system (e.g., the atmosphere, carbon in the atmosphere)
- **Flow**=The rate at which something flows between components
  - Inflow=Rate of flow of something (e.g., energy, matter, organisms) into a stock
  - Outflow=Rate of flow out of a stock
- **Interactions or processes** = The ways in which components interact with each other
- **Feedbacks**=One of the ways that interacting components or stocks produce behavior in the system, can be balancing feedback or reinforcing feedback.

Teacher Note

Teachers should keep the systems terminology and definitions displayed in the classroom during the unit.

Activity 2: The Bathtub model

Students are introduced to the Bathtub Model, in which the bathtub represents the atmosphere, and carbon emissions are the water filling the tub:



The Carbon Bathtub and its Components

Source: Causes of Climate Change: Climate Change: US EPA

In pairs, students should change the variables in the interactive bathtub model at:
https://www.climateinteractive.org/tools/climate-bathtub-simulation/

To complete the activity, they discuss and write down what they have learned as follows:

1. What is represented by:
    a. The bathtub
    b. The water
    c. The flow rate of water entering the bathtub
    d. The flow rate of water leaving the bathtub
    e. The level of water in the bathtub.
2. Why it is not enough to level off the rate of emissions (flow from faucet into the bathtub).
3. How the model falls short in representing the current imbalance in Earth's carbon cycle.

The teacher has students share their responses in a whole-class discussion, particularly responses to why the tub overflows even if emissions level off, and how the analogy falls short as a model.

Teacher Note

1. Even if emissions level off, the level of water is still higher than the outflow from the tub through the "drain." Emissions will need to be reduced to the point where they start declining in order to prevent the tub from overflowing.

This is because there's a delay in temperature increase as the climate "catches up" with all the carbon that's in the atmosphere which will only stabilize about 4 decades later. This decades-long lag between cause and effect is due to the long time it takes to heat the ocean's huge mass. The "extra" energy held in the Earth system by the increased carbon dioxide not only heat the air but also melts ice, and heats the ocean. Compared to air, it's harder to raise the temperature of water –and so it takes longer. However, once the ocean temperature is elevated, it adds to the warming of the Earth's surface. Scientists estimate that this "catch up" commits us to another approx. 1 degree F after emissions stop completely.

2. How does the model fall short as an explanatory model? The model falls short because there is more than one inflow into and more than one outflow from the atmosphere.

Activity 3: Revisit the systems diagram

With the systems diagram they generated earlier displayed, students review the components and interactions (arrows) represented there. The teacher emphasizes that this is a systems representation, and connects it to the systems terminology that has just been introduced, by asking students to identify the stocks and flows in the diagram.

Next, the teacher asks them to make one change to the diagram, e.g., take something away, modify a connection or add something, and asks them how they think this would impact the behavior of the system.

Teacher Note

Each version of the systems diagram can be photographed and saved for reference and to demonstrate progress.

Exit ticket

Students write: Why do you think that changing one component in the system diagram would have an impact on the system's behavior?

Section 3: More about climate change

(3 class periods)

Overview

In this section, students review the topic overviews in the Learning Library. The teacher chooses one of three learning models to actively engage students with the material.

Learning goals

- Students will be able to teach, and learn from, their classmates some important concepts relevant to climate change.

Do Now

Class period 1: In their notebooks, students write responses to the following question:

- List what you believe are the 3-5 most important causes of climate change that people should know about.

Class period 2: In their notebooks, students write responses to the following question:

- List what you believe are the 3-5 most important effects of climate change that people should know about.

vClass period 3: In their notebooks, students write responses to the following question:

- List what you believe are the 3-5 best ways to mitigate (lessen) climate change that people should know about.

Activity 1: The Learning Library

The project library online provides an overview of each topic relevant to climate change, together with related animations, visualizations, videos, and games. First, the teacher introduces the idea of the library, then chooses to organize students in one of three ways to engage with the content of the overviews, depending on the particular needs of the classroom. It is expected that in this section, students will first engage with the content in the topic overview.

Later, as they develop their games, you will need to encourage them to consult the animations, videos, etc., so that they can deepen the science content of their games.

Teacher Note: *These resources will be the primary ones that students draw on for content for their games. Students consult them for inspiration, and for data to incorporate into their games. The topics in the resources are:*

- *A systems approach to Earth's energy balance (radiation and re-radiation, albedo, the greenhouse effect)*
- *Earth systems*
- *How Earth's subsystems (atmosphere, biosphere, hydrosphere, etc.) interact*
- *Anthropogenic disruptions of the carbon cycle, including climate change as an accumulation problem*
- *Impacts of climate change*
- *Benefits of taking action.*

1. STUDENT PRESENTATIONS

Student pairs take on a topic and create a three-minute presentation - based on the topic overview - to share with the class, using these guidelines:

How to give a three-minute talk: You and a classmate are going to create a three-minute speech about all the topics in the learning library that are related to climate change. The goal is to share this task so that you get a quick overview without having to read all the topic overviews. This will help you and your partner to choose what topic your game will focus on. After that you will be able to explore your chosen topic in more depth as you design your game.

Here are some steps:
1. Find the Main Idea: Identify the main idea in the topic overview – your presentation should have *one* message. Next, identify 2-3 ideas or topics related to the main one.
2. Get Organized: Organize the main topic and supporting ideas into a structure:

- Introduction – the main topic
- Body – 2-3 ideas related to the main topic
- Conclusion – repeat the main idea.

3. Get Ready to Present:

- Read it through at least 3 times, so you're reasonably familiar with what you want to say. Then you'll be able to look at your audience when you talk to them.
- Try to vary your tone of voice – monotones put people to sleep!
- Speak clearly, and avoid "ums" and "like" – these are fine in everyday speech but distracting to your audience.

2. JIGSAW METHOD

Divide the overviews in the learning library into roughly 6 parts. Assign students into "home groups" of 6 students each. Give the students in each group a number from 1-6. All #1 students read and make notes on the same part of the material, all the #2 students a different part and so on. Once they have read the material, the "experts" on each part get together to discuss the main points of the reading and plan a strategy for teaching this material to their home group. Experts return to their home groups to teach the topic, so that each group hears about all the topics.

3. GALLERY WALK

Assign student pairs and assign each pair a topic from the learning library. Have them create a visual display of their assigned topic to share with the whole class. Students circulate to view and take notes from the displays.

Activity 2: Refining the systems diagram

Students wrap up this section by returning to their systems diagram. In their original small groups, they add new components based on what they have learned from their work during this section. This will give the teacher some formative feedback, which will inform the extent to which students will need to return to the learning library in Section 6 when they finally come to their game design.

Activity 3: From systems diagram to connection circle

So far, student systems diagrams have followed a traditional concept map design, which makes it easier for students to "tell the story" of the system to construct the diagram. Now that they are familiar with more climate complexity, a connection circle will help them make the conceptual jump to more abstract **systems thinking**. This will help them think about all the **causal connections** among the systems components and also see the **feedbacks**. Students draw a circle on a large piece of paper (poster paper), and write all the relevant climate components around the outside of the circle. They draw arrows to try to represent all the connections that they listed.

Teacher Note

Guidelines for creating and working with a Connection Circle:

- *Choose components of the system that:*
  - *Are important to changes in the system*
  - *Are **nouns** or **noun phrases***
  - *Increase or decrease in the story*
- *Write the components around the outside of the circle*
- *Identify components that cause other components to increase or decrease, and draw an arrow from the cause to the effect (e.g., an arrow **from** burning fossil fuel **to** greenhouse gases, or **from** higher ocean temperatures **to** melting sea ice)*
- *Look for feedback loops (e.g., less sea ice, to lower albedo, to higher solar energy absorbed, to less ice)*

Learn about teaching with connection circles at: https://thesystemsthinker.com/learning-about-connection-circles/

Exit tickets

Class period 1: List 3-5 new things that you learned about climate change.
Class period 2: Sketch the carbon cycle as a system of stocks and flows.
Class period 3: What did you add to or change in your systems diagram, and why?

MODULE TWO
Section 4: What are games?

(1 class period)

Overview

Students compile a list of games, categorize them based upon features, and then contrast the different genres. They will play and critique a selected sample of online games related to climate change in order to identify features that make a "good" game, and that contribute to learning.

Learning goals

- Students will be able to classify different types or genres of games by identifying their characteristics.
- Students will be able to describe the characteristics of educational games that help them support learning.

Do Now

Students write a list of examples of games (online or in the real world) they play now, or have played before.

Activity 1: What are games?

In small groups, students share their lists of examples of games they play or have played, both physical and online. As a group, they create categories to sort the games into, based upon features the games have, game genres, or other criteria the games have in common (there isn't one ideal categorization system). Afterward, they will report out to the class their categories and rationale.

> ***Teacher Note:*** *Examples of categories students come up with might include:*
> - *Quest games, e.g., Hero simulator, Ruthless Pandas*
> - *Sports games, e.g., World Cup Soccer, Ultimate Dodgeball*
> - *"Story" Games in which characters, e.g., Adventure Story, Toy Story*
> - *"Educational" games, e.g., Zoombinis, Quantum of Light*
> - *Puzzle games, e.g., 10, Tetris*
> - *Action games, e.g., Slither.io, Zombo Buster Rising*
> - *Strategy games, e.g., Kingdom Rush, StarCraft.*

As they report out, students should be prompted to brainstorm and record features of the games on their lists:

- What are the features that the games share?

- How do those features relate to each other?

As multiple groups share their categories, students should consider how the same features may relate to each other differently in two different games. For example, both "Go Fish" and "Poker" include playing cards and players, yet are very different games. Students may also compare how different groups may have categorized some of the same games differently, despite having the same features. For example, "Wii Sports" might reasonably be described as a sports game, a physical (movement) game, or a simulation game. The discussion of how game features relate to each other will support students to think about games as systems, in the next activity.

Activity 2: What are good games?

The goal of this activity is to get students to begin to think about games as systems. What are the components of games? How do they interact? How do actions in the game affect how future play unfolds? For example, some games, like Pacman, include "enemies." Simply increasing or decreasing the number of these components drastically changes the difficulty of a game. As another example, many games include similar physical materials (e.g. card games). However, the relationships among those elements in the game (i.e. the game rules) make each a different game.

Different genres of games also represent different kinds of systems. A very **simple puzzle or quiz game** doesn't include many relationships between its components—the player doesn't do very much. As a result, the system is static. A game that allows players to earn or trade resources, or build abilities, may construct a complex system in which a player can interact. Talking about game genres also gives students an opportunity to think about what kinds of games they find engaging or boring (and why). These considerations will all shape their thinking about what it means to create an engaging game that will also educate – not an easy task!

Students think about which of the features they've listed make for a "good game." Next, they explore the learning opportunities inherent in the games we've provided, by looking for what features make for engaging games that educate, and by looking for what not to do. They play a sample of the educational games about climate change provided and critique them. The set will include deliberately bad games for contrast (but the content will be sound).

The games can be divided up jigsaw style so that not all students play all games. As they play, they should answer the following questions (can be provided as a worksheet or list of questions on the board):

- What makes the game work? (What are the parts? How do they work together?)
- Is the game easy? Hard?
- Does the game include a system? If so, what system?
- What can you learn from the game?

Students share out at the end of the session, and revise their list of features of what constitutes a good game.

Teacher Note

1. Ideas students might come up with include:

- *level of challenge (not too hard, not too easy)*
- *fun to play (but, what makes it fun?)*
- *it lets the player win ("I don't like games where it is too difficult to win, because you get frustrated"*
- *looks realistic*
- *it lets you play with other people*
- *contains strategy ("You can be good at the game, it's not just luck")*
- *is fair (the rules don't preference one player over another)*
- *it lets the player make interesting choices*
- *different every time you play, etc.*

They may or may not include "lets you learn things" in their list of ideas.

2. Students might be prompted by examples of horrible games, for example, imagine a game called Heads or Tails. One player flips a coin, while the other guesses, and then finds out if they're right. This would be a boring game!

Game examples:

1. Games designed by the project:

Albedo: https://scratch.mit.edu/projects/118677306/
Insulation: https://scratch.mit.edu/projects/118691765/
Clouds: https://scratch.mit.edu/projects/118693595/

2. Scratch games by students:

Arcade (simulation) games:

- Dodging/reflecting
  Greenhouse Guardian: https://scratch.mit.edu/projects/107590617/
  MaddieMaeve Climate systems project: https://scratch.mit.edu/projects/164005110/

- Clicking
  Scratchy climate change: https://scratch.mit.edu/projects/108549906/
  Factory game: https://scratch.mit.edu/projects/24324063/

- Actual simulation (like "Civilization" game)
  Evergreen 2g Science project game: https://scratch.mit.edu/projects/165086014/
  Evergreen2b game: https://scratch.mit.edu/projects/165085276/

Narrative/choice games:
A&J game: https://scratch.mit.edu/projects/108355966/
Comfortable climate: https://scratch.mit.edu/projects/165084615/

2 player games:
Xfluffles Climate change game: https://scratch.mit.edu/projects/108549966/

Sage3ebg Climate systems game: https://scratch.mit.edu/projects/163989999/
Evergreen2g Science project game: https://scratch.mit.edu/projects/165086014/

Platformer game:
Sage1kj climate systems game: https://scratch.mit.edu/projects/164006163/

3. NASA games:
Offset: http://climatekids.nasa.gov/offset/
Powerup: http://climatekids.nasa.gov/power-up/

Teacher Note

We recommend that you explicitly "ban" students from designing quiz or maze games, given that they offer very poor options for representing interactions in systems. Here are two such games:

Climate jahooty: https://scratch.mit.edu/projects/164008615/
Dinoscratchlover climate game: https://scratch.mit.edu/projects/108355543/

Exit ticket

Students list examples of one game that includes a system, and one that doesn't, and explain.

Section 5: Designing in Scratch

(3 class periods)

Overview

In this section, students will become familiar with Scratch, or refresh their memories if they have created in Scratch before. They will engage with Scratch activities appropriate to their level of experience with Scratch. **Note: If you haven't already, give students the Experience Survey so that you can appropriately identify suitable student pairings for game design for the rest of the unit.**

Learning goals

- Students will be able to create a project in Scratch that includes a system.

Do Now

Class period 1: What is important to you in a game? Can you give an example?
Class period.2: What game did you learn something in? What did you learn? Why?
Class period.3: What is a bad game that you have played? What made it bad?

Activity 1: Designing in Scratch

The teacher should read through Unit 0, 1 and 4 in the Creative Computing Guide to become familiar with the three levels of Scratch activities presented below. Also, students need to create accounts, and the teacher needs to set up a studio for students' games.

Teacher Note

1. Setting up a Scratch account: Student handout has instructions.

2. Creating a Studio: In your account, click on your profile button. Select "my stuff" from the dropdown. Select "add studio" and name it. You may wish to have a studio for each class, eventually. On the other hand, you might also want all the projects to be together so students can find each other's work easily.

3. Some of the videos for this Activity are on youtube. If this website is blocked in your school, you will need to copy pages from the Creative Computing Guide to introduce the user interface to beginners.

4. "Unit 4: Games" in the Creative Computing Guide has several important and useful tools that will help students with Activity 3, and with their game design later.

BEGINNER: GETTING TO KNOW SCRATCH

Get started

(https://sites.google.com/site/scratchadventures/getting-started)

- ☐ Go to the Scratch website and join. While waiting for others, have students explore and play games.

- ☐ Watch the User Interface (6:26 minutes) video to learn about:
  - Stage
  - Sprite List
  - Blocks Palette
  - Scripts Area
  - Cursor Tools

Move the cat

(https://sites.google.com/site/scratchadventures/lessons/1-move-the-cat)

- ☐ Watch videos: Project Page (4:53 minutes)
- ☐ Create your first project by following the Scratch Getting Started Guide.
- ☐ Share your project.

Sprites

(https://sites.google.com/site/scratchadventures/lessons/sprite)

- ☐ Watch Paint Editor (9:14 minutes) and Sprite 1 (4:18 minutes) videos.
- ☐ Go to your 'my stuff' Scratch page and create a new project.
- ☐ Draw your own sprite and make it move.
- ☐ Select a sprite from the Library. Click on Costume and then change the color using the paint editor.

Loops

(https://sites.google.com/site/scratchadventures/lessons/loops)

- ☐ Watch Loops (2:17 minutes) and Events (2:08 minutes) videos.
- ☐ Remix Loops project and try out the exercises shown in the video.

Other lessons to consider:

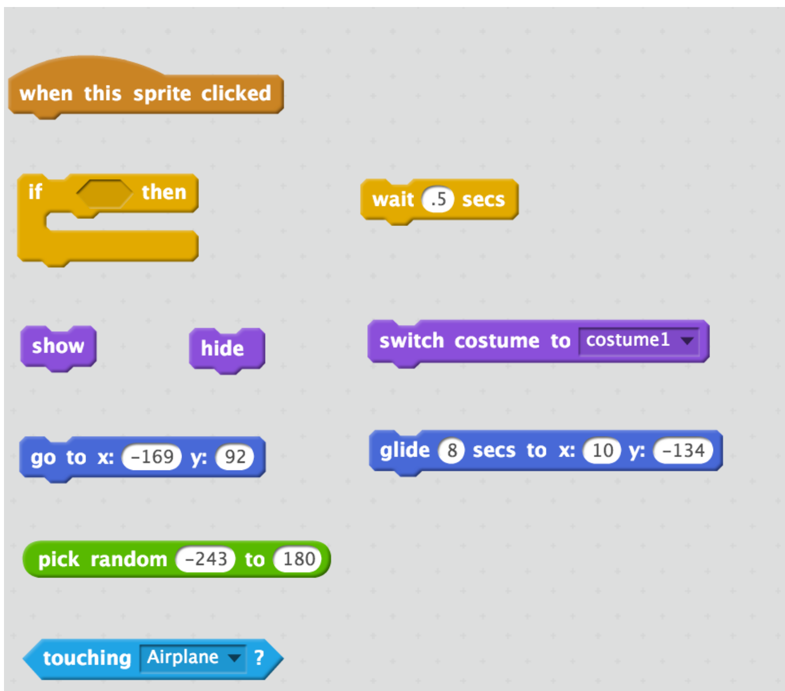
- Motion
- Looks
- Debug
- Costumes
- Backdrop
- Create studios
- Pen.

INTERMEDIATE: THE 10 BLOCK CHALLENGE

Students with some Scratch experience do the 10-block challenge. They must create a project by remixing, using only these 10 blocks (shown in the screen below). They can use them once, twice, or multiple times, but must use each block at least once. Project the blocks in the classroom, or provide them with the illustration below.

Teacher Note

A remix is a piece of media which has been altered from its original state by adding, removing, and/or changing pieces of the item. A song, piece of artwork, book, video, or photograph can all be remixes. The only characteristic of a remix is that it appropriates and changes other materials to create something new (Wikipedia). Remixing is a good strategy for programmers - students included. Some students may not think of it, or may consider it "cheating." Students use backpacking to copy and import code from other projects.



EXPERIENCED: LEARN SOME MORE ADVANCED FUNCTIONS

The following resources, created by the ScratchEd team, provide examples of projects that are based on more advanced functions, some of which students may find useful as they code their games later. Now, students can interact with the projects and then "see inside" to the scripts. Later, they can backpack the scripts into their own projects. Once in their own projects, they will be able to remix the scripts to suit their needs.

How to change levels: http://scratch.mit.edu/projects/1940453

How to program the mouse to control play: (http://scratch.mit.edu/projects/25192659)

How to create and use variables: (http://scratch.mit.edu/projects/1940456)

How to make a block: (https://sites.google.com/site/scratchadventures/scratch-studios/make-a-block)

'Make a Block' provides the ability to reuse the same code in multiple places without having to duplicate the code.

Activity 2: Debugging challenge

Students are presented with several debugging challenges. Working to fix the code will present them with several problem-solving challenges, and also help them learn new functions. The challenges can be found at:

https://scratch.mit.edu/studios/2034712/

All student pairs should do the first one, before moving on to other challenges.

> ***Teacher Note***
>
> *Debugging tip: Ask students to read their code aloud, similar to how you would read a sentence, to a student partner. Often times, errors or missing pieces of code will become obvious to either the reader or the listener while it is being read aloud.*

Activity 3: Systems in Scratch

In activity 3, all students modify an existing game to create a system. Students should choose one of the Scratch games designed by students that they played in Section 4.

But first: A system has parts (components) that interact. An example of an interaction in Scratch includes a sprite hitting something, or running towards or away from something. However, systems require interacting parts that change one or the other or both of the agent's sprites. These changes could be temporary (a sprite saying "excuse me" when it bumps into another) or permanent (a sprite disappears when it bumps into another). Therefore, to create a system, students will need to add **"sensing" and/or "interacting" and/or "broadcasting"** functions to their games.

1. One example of a fairly **simple** system can be found in the following game:

https://scratch.mit.edu/projects/119235434/

Project the game for the whole class to see, and ask for a student volunteer to play the game.

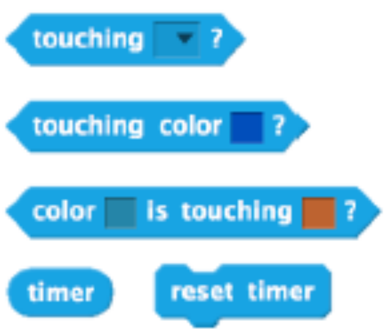
2. Students should discuss what interactions in the game might constitute a system. (In this game, the player uses a CO2 molecule to trap solar radiation from escaping into space, thus making the temperature go up. System components: Earth, CO2, solar radiation, temperature, time (the solar radiation speeds up) - and possibly humans. The temperature level triggers the "you win" screen when it reaches a certain "threshold.")

3. Next click the "See inside" tab to project the code. Ask for a student volunteer to explain how the code results in the onscreen behavior.

4. Ask students to suggest what component and interaction might be added to the game to make the system more complex.

5. At this point you may want to convene mini-lessons about two Scratch functions, sensing and variables, that help students create systems. Ask student volunteers who know how to do these already to teach a small group. If there are no volunteers, consult the scratch wiki for how-to (**https://en.scratch-wiki.info/Sensing_Blocks)**. The suggested mini-lessons are:

a. Sensing
The following sensing blocks are useful.



b. Variables.
To create a system, students may also want to use variables. Variables are inherent in games. They help with keeping score and establishing thresholds. Operators (see Teacher Note below) can be used to count or manipulate variables. The following blocks will help students to create systems in their projects:

Create Variable, Set Variable, Change Variable
https://wiki.scratch.mit.edu/wiki/Variables_Tutorial

Check out this project to see how to create a score: **https://scratch.mit.edu/projects/2042755**

Cloud Variable

Cloud Variables allow the designer to make a high score that can be stored rather than starting high score at zero every time.

c. Creating thresholds (either for system behavior, e.g., sea ice melting once a certain temperature is reached, or for game behavior (e.g., game ends when CO2 reaches certain level).
Stress to students that an **interaction doesn't necessarily constitute a system on its own – a system does not always change just because two parts interact.** There has to be a threshold. For instance, clouds can only hold so much water vapor before the vapor gets together, forms droplets and turns to rain. To create this scenario, a designer could have droplets bounce around in a cloud by using a "glide to x and y" block and inserting the "random operator block" in the circles for X and Y coordinates in the glide block.

The designer would also use the "Create Variable Block" to create a variable that would represent the weight/size of the water droplets.

- variables allow states of sprites to possibly change when the variable reaches a certain threshold (change variable by X block).
- They also work as point counters for keeping score.

The droplets would also have "sensing blocks" that changed the size of the water droplets when they touched another droplet sprite. When the droplets touched each other, they would also activate the "change the variable" block by 1. When the variable reached 30, it could then send a "broadcast" and create rain by having new sprites receive the broadcast, show, and glide down from the cloud. In this example, the water droplets stick together once they touch:

https://scratch.mit.edu/projects/119239124/

6. Next, have students modify an existing game to create a system or, if the game they choose already includes a system interaction, have them add another system component. They should choose one of the Scratch games designed by students that they played in Section 4:

Arcade (simulation) games:
- Dodging/reflecting
Greenhouse Guardian: https://scratch.mit.edu/projects/107590617/
MaddieMaeve Climate systems project: https://scratch.mit.edu/projects/164005110/

- Clicking
Scratchy climate change: https://scratch.mit.edu/projects/108549906/
Factory game: https://scratch.mit.edu/projects/24324063/

- Actual simulation (like "Civilization" game)
Evergreen 2g Science project game: https://scratch.mit.edu/projects/165086014/
Evergreen2b game: https://scratch.mit.edu/projects/165085276/

Narrative/choice games:
A&J game: https://scratch.mit.edu/projects/108355966/
Comfortable climate: https://scratch.mit.edu/projects/165084615/

2 player games:
Xfluffles Climate change game: https://scratch.mit.edu/projects/108549966/
Sage3ebg Climate systems game: https://scratch.mit.edu/projects/163989999/
Evergreen2g Science project game: https://scratch.mit.edu/projects/165086014/

Platformer game:
Sage1kj climate systems game: https://scratch.mit.edu/projects/164006163/

Teacher Note

Read the section in Unit 4: Games in the Creative Computing Teacher Guide, p. 84-85 on interactions. This unit will familiarize you with several important Scratch tools which you might want to keep in mind as students begin to want such features as they get into game coding. Examples given illustrate interactions, scoring, and other more advanced gaming tools like levels, timers, and 2-player games. Note that students can take responsibility for finding these by asking others, or by googling, but it is also good to have them in your back pocket as possible suggestions if they get stuck.

Useful gaming tools include the following:

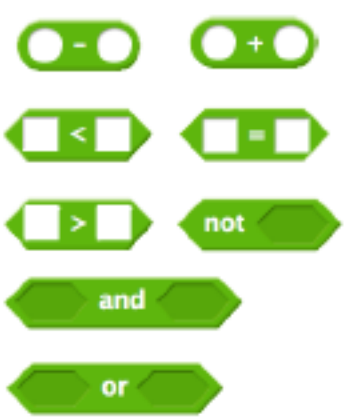
Lists

*A **list** (also called an array in other programming languages) is a tool that can be used to store multiple pieces of information at once. It can also be defined as a variable containing multiple other variables.*

https://wiki.scratch.mit.edu/wiki/List

"Operators"

Operators set quantities, or amounts, that can allow sprites to change when a certain variable amount has been reached. The following are useful operators: ***+ - * / X = Y***



Pick Random

If/then/else *– blocks allow for sprites to operate normally until they meet a certain criterion – like touching another block.*

Broadcast (send and/or receive)

Allow multiple sprites to react in their own way when a certain state is achieved.

Exit ticket

Class period 1: Describe two new things you learned to do in Scratch.
Class period 2: List two things you learned from debugging.3: Describe two ways to create a system in Scratch.

Teacher Note

Extensive resources to support the Scratch activities can be found online. A community has grown around the use of Scratch in the classroom, and resources and tips suggested by this teacher community are available at:
http://scratched.gse.harvard.edu

Among the resources on this site is an extensive ***Creative Computing Teacher Guide****, which you should download and familiarize yourself with to the extent that you can point students to particular resources if they get stuck. It can be found at:*
http://scratched.gse.harvard.edu/guide/

There is also a handy spreadsheet created by the folks at Scratched that can be quickly and easily scanned to locate a particular resource:
https://goo.gl/EWbzVz

MODULE THREE
Section 6: What's my game?
(1-2 class periods)

Overview

Students choose the topic for the game they are going to design, and use a design sketch template to plan it.

Learning goals

- Students will be able to use triadic game design – focusing on Reality, Meaning and Play – to create design sketches for their games.

Do Now

Students write what specific climate change content they will want the player learn while playing their game.

Activity 1: What's my game?

Students now have some basic information about climate change and Systems thinking from Sections 2 and 3, some understanding of the basics of game design from Section 4, and some Scratch skills from Section 5. Now, students turn to designing their own games. First, they must choose the topic of their game. Working in pairs, they choose one of the categories listed below, and brainstorm several possible topics in the category (the categories roughly match the sections in the learning library). They also determine what they think they will want the player to learn. Categories:

- Earth's energy balance (radiation and re-radiation, albedo, the greenhouse effect) and global warming
- How Earth's subsystems (atmosphere, biosphere, hydrosphere, etc.) interact
- Anthropogenic disruptions of the carbon cycle
- Mitigating impacts of climate change, e.g., drought, heat.
- Reducing emissions (e.g., CO2 capture, reducing meat-eating, planting forests)
- Benefits of taking action.

The following requirements MUST be met:

1. The game must relate to some aspect of climate change
2. Designers must consider:

-Reality (how can we accurately represent climate change)

-Meaning (what will the player learn – the goal of the game)

-Play (what will be the look and feel of game play on screen)

3. Include a system.
4. Include a human connection.

In conjunction with describing the requirements for the game, the teacher introduces the **Reality cards**, which are designed as possible prompts to brainstorming. Optionally, the teacher can assign these to the pairs at random so that they must take on the topic on the card as the Reality aspect of their game. These can be found on the Building systems from Scratch website:

https://www.buildingsystems.terc.edu/

Students describe the Reality topic using the Design Template at the end of the lesson (and in the Handouts and Worksheets section).

Activity 2: Design sketching

When students have chosen the reality aspect of their game, and are ready to begin design sketching the game, the teacher should hand out the **Play and Meaning Cards**. These should be assigned to pairs at random. Students should work with the cards they've been given – research has shown that this increases creativity (but see Teacher Note below).

Students use the provided Reality, Meaning, Play template (see next page) to begin design sketching their proposed game. They can use the space provided to write about and/or create sketches of their game ideas. The teacher can circulate to support and encourage students.

Examples of scaffolding questions the teacher can ask them as they get started:

- What will be the look and feel of your game?
- What is the system in your game? (What are the parts of your game? How do they interact?)
- What are the characters? The setting?
- What do you want the player to learn from your game?
- If you were to explain to someone how to design your game, what steps would you tell them?

Teacher Note

1. If students are really stuck with the Play and Meaning cards they've been assigned, they can switch to another, but only once.

2. Remember that certain game genres do not lend themselves to game play that focuses on systems. Students may opt for designing quiz games because if they've played games in school, they were most likely quiz games, and so this genre is what they think of as an "educational game." A very simple puzzle or quiz game doesn't include many relationships among its components—the player doesn't do very much in playing the game. As a result, the system is static. The teacher should definitely exclude quiz games from the list of possibilities, and probably puzzle and maze games, unless they are not simple puzzles or mazes.

3. The majority of students will be creating a subsystem and not the entire climate change system. The entire system would be a programming challenge beyond the expertise of most students, so be on the lookout for students who might be planning to do this. Check with them what their level of expertise is, and guide them to a manageable part that they could take on instead.

Exit ticket

Students write a sentence or two to describe what system they plan to include in their game, and how it's parts will interact.

REALITY, MEANING, PLAY TEMPLATE

1. REALITY: Sketching your climate change topic

Topic What will be the overall climate change topic of your game? What specific topics do you want it to be about? How will *humans* act in your game?
System What parts of what system will you design in your game, and how do they interact with each other? Draw a systems diagram to show the system. Try to include at least one feedback in your system.

2. MEANING: Sketching the learning goal of your game for the player

Purpose What do you want the player to learn from the game?
Plan How will your game achieve this learning goal?
Values Are there values or moral beliefs about climate change you want to represent in your game? What ones? How?

PLAY: Sketching the genre of your game

Overall Goal What will the player do in your game (*e.g. reduce emissions, save planet earth, take CO_2 out of the atmosphere*)?
Gameplay What will be the rules of your game? What actions will players perform?
Game world What will be the story in your game? How will your game space look and feel on the screen? (*e.g. colors, sound, realistic or not?)*

Section 7: Game design to learn the details of climate change

(10+ class periods)

Overview

This is when students get into serious game design. They now transfer the ideas they have detailed in their design sketches into Scratch by engaging in systematic (algorithmic) design steps that include rapid cycles of planning, designing, playtesting.

Teacher Note

1. *You may find it useful to review Unit 4: Games, in the Creative Computing Teacher Guide, before your students begin this section.*
2. *Review a video on **pair programming** at: https://code.org/educate/resources/videos (2:50 min). Scroll to "The basics of computer science" to find the video. Although the video may seem a little "silly," it is worthwhile showing, to outline roles that work well for pair programming. Students take turns as "Driver" in charge of the keyboard, and as "Navigator," making suggestions, pointing out potential problems, etc. It outlines rules (at the 2:39 mark) which you can post in the classroom:*

DO: Be respectful
DO: Talk to one another about the work
DO: Explain what you are doing
DO: Think ahead and make suggestions
DO: Switch roles often
DON'T: be a bossy navigator
DON'T: Grab the driver's mouse/keyboard

The video suggests that you set a timer and insist upon students switching every 10 minutes when they are programming to accustom them to this pair programming practice. You can relax the timer rule once you see students being generally collaborative.

3. *Encourage students to "remix," using blocks of code from other projects. Remind them that it is good programming practice, not cheating. Remixing obviously reduces the necessity for creating code from scratch. (If questions about plagiarism arise, take the opportunity to discuss: http://scratch.mit.edu/help/faq/#remix.)*
4. *If students work on their games outside of class, warn them to plan with each other about who and when. This is important because the project becomes "locked" by the student who opens it, with the result that none of the edits of the other student gets saved. One way to avoid this happening is for them both to choose the "remix" option when they open the file, each create a new file name to save their work, then share their new work the next day in class via the backpack feature.*

Learning goals

- Students will be able to transfer their game ideas into Scratch programming by engaging in systematic and algorithmic design (iterating through planning, designing, playtesting steps).
- Students will be able to represent some aspect of climate change accurately in their game.
- Students will be able to explain the intended learning goals of their game.
- Students will be able to demonstrate computational thinking skills.
- Students will be able to give and receive constructive critiques.

Teacher Note

We suggest the following sequence of activities for this section. ***(Note that an exit ticket should be done at the end of each lesson.)***

Class period 7.1
- *Revisit games (Activity 1)*
- *Design (Activity 2)*

Class period 7.2
- *Design (Activity 2)*

Class period 7.3
- *Design (Activity 2)*
- *Critique (Activity 5)*
- *Revisit systems diagram (Activity 4)*

Class period 7.4, 7.5, and 7.6
- *Design (Activity 2)*

Class period 7.7
- *Design (Activity 2)*
- *Critique (Activity 5)*
- *Write reflection on game (Activity 6)*

Class period 7.8 and 7.9
- *Design (Activity 2)*

Class period 7.10+
- *Design (Activity 2)*
- *Critique (Activity 5)*

Do Now

Class period 1: Read about your game topic in the Learning Library.
Class period 7.2: Describe how remixing could help you code your game.

Class period 7.3: Revisit and review your design sketch to make sure that you are including the components you originally intended to include. Write down what you don't know how to program to create those components.
Class period 7.4: Write about what and how the player will learn from your game by playing it.
Class period 7.5: Draw a systems diagram to represent the climate change in your game.
Class period 7.6: Describe one system you have included in your game.
Class period 7.7: List one new climate change aspect you could add to your game.
Class period 7.8: Describe one way in which you and your partner solved a problem in creating code for your game.
Class period 7.9: You think your game is complete. BUT, write one way in which you could add another system to your game.
Class period 7.10: If you were expert, what is one additional thing you would have liked to have in your game.

Activity 1: Revisit some example games (optional)

Since at least a week and a half have elapsed since students played and critiqued the sample games provided, you may want to refresh their memories. You will need to decide if they need to play a couple of the games again, or whether it will be sufficient to remind them of the list of game genres they came up with, as well as the list of game design features that constituted a good game.

Activity 2: Design!

Student pairs now transfer the ideas they have detailed in their design sketches into Scratch. Examples of possible types of games we think students might be interested in developing include:

- Carbon dioxide removal games (e.g., through tree planting)
- Social games (e.g., strategies for encouraging people to save energy)
- Engineering games (e.g., deploying renewable energy, or sustainable agriculture)
- Climate modeling games (e.g., player manipulates the carbon cycle)
- Mitigation games (e.g., through reducing emissions).

Teacher Note

1. As they begin to create code, students may find that their original ideas are not feasible in Scratch, or that they do not have the necessary expertise to program their ideas. This is to be expected. However, before they abandon these ideas, they should:
i) consult with the Scratch "experts" in the classroom
ii) review the Scratch binder of programming tips
iii) explore the scratch.mit.edu site to see if they can find a game similar to their planned genre, from which they might remix the code
iv) look for a solution online by googling their question – there are many short videos made by students that explain how to solve specific design/coding problems.

2. NASA has a library of climate change related images that students may find useful at: http://climatekids.nasa.gov/menu/teach/
Also, students may find images to use at https://scratch.mit.edu/studios/1671324/projects/

When appropriate, for example when students have made a solid start on their programming, the teacher introduces the Rubric for game design (provided in the Blackline masters). Tell students that they should use the rubric as an additional guide for designing an excellent game, and that you will be using the rubric to assign part of their final grade.

Activity 3: Mini-lessons

You many notice that some students could use a little extra support during this section, either with programming or with particular climate change content. If this is the case, the teacher might facilitate a "mini-lesson" in one corner of the room. Without distracting the rest of the class, the teacher can bring a small group of students together to quickly problem-solve or to explore a topic in more depth, with the teacher's support as/if needed. The "help ticket" area (see Activity 7) can provide input about the need for topics for mini-lessons.

Activity 4: The systems diagram, again

After about 3 class periods during this section, students should be re-assigned to their original systems diagram groups (Section 1), and add to the diagram to incorporate new content they have learned during game design. Remember, this is an important step to support the learning of all students, since they will not be incorporating every aspect of climate change and systems into their individual games.

Teacher Note

1. At this point, encourage students to annotate any feedbacks they have identified, either balancing or reinforcing, by annotating the relevant arrows/variables.
2. Older versions of the diagrams can be photographed and saved for reference and to demonstrate progress.

Activity 5: Constructive critique

Once they've worked on introducing a system into a Scratch project, students will critique the project. The goal is to learn how to provide constructive feedback, push designers to go deeper on climate change concepts, and "up" the coding level of their games.

Students critique the Scratch project of another pair, learning how to use a simple critique framework (the form is provided at the end of this section). If a pair wants particular help, they should identify that so that their peers (or teacher) can address their request.

Teacher Note

1. The Critique Form can be found at the end of this section. As an important part of good coding practice, critique also provides a valuable opportunity for teacher and classmates to provide feedback that helps make for deeper climate change concepts (Reality), better alignment with the intended learning goals for the game (Meaning), and for improving the design (Play). If you suspect that your students will not take this task seriously, you could assign a portion of a grade for the quality of their critiques.
2. In addition to this formal critique, note that students will also engage in spontaneous critique. The teacher can expect students to want to play other students' games, to try to "break" their games, and to provide their own critiques. This is taken seriously by the vast majority of students in a participatory culture, proves to be a highly valuable spontaneous classroom activity, and should be encouraged!

Activity 6: Written reflection

Each student writes a reflection about their game, in response to the following questions:
1. What is the climate change science the player learns from my game?
2. How does the player learn the science? Be specific.
3. Why did you design the game the way you did?
4. What is the best feature of your game?
5. What specifically needs to be improved?

Activity 7: Exit tickets

At the end of each class period during this entire Section, student pairs should take 3 minutes to complete an exit ticket. Not only will this provide the teacher with some idea about this aspect of their computational thinking, but it also provides students with a valuable opportunity to reflect on their process, and place an emphasis on explicitness and systematics thinking, both tools of good programming.

An alternative, more public, way to determine progress would be to dedicate a scratchpad or a section of the whiteboard as the "Help ticket" area. Students could write up the action they don't know how to design, that others who do know can help them with. The teacher will periodically need to draw attention to the status of the help ticket.

Exit ticket questions:

- What are we struggling with? What did we do?
- What do we need to do next?

"Our game is done!"

Some students may declare that their game is done after just one or two days of game design. You may find it useful to require that students have you sign off on their games – this will give you the opportunity to suggest ways they can extend themselves further.

Teacher Note

Examples of ways to extend the systems aspects of games:

In a game that has one causal relationship between two variables, such as cars generating emissions, suggest that students add a third variable such as a thermometer that measures increase in temperature.

In a game that has causal relationships among three variables, such as cars generating emissions which increase temperature, suggest that students add a feedback. For example, at increased temperatures cars generate more emissions (because they are less efficient at higher temperatures) which reinforces the feedback to increased emissions and increased temperature rise.

In a game that includes a decorative element in their game that doesn't have any function in gameplay, suggest that they try to think of ways to incorporate the element into gameplay. For example, decorative snowflakes in a game based in the Arctic could have a causal relationship to the warming happening in the rest of the game.

CRITIQUE FORM

Feedback from: ____

Feedback for: ____

	What **works well or you really like?**	What is **confusing?**	What could be **improved?**
PLAY: How engaging is the game? How did you feel as you interacted with it?			
MEANING: Is the learning goal obvious to you?			
REALITY: Is the climate change topic clear? Can you identify the system in the game?			

Section 8: Completing and showcasing games
(2 class periods)

Overview

In this section, students will complete their games, showcase them for an appropriate audience.

Learning goals

- Students will be able to communicate about global warming through presenting their games.

Do Now

Students identify a task or tasks they still need to do.

Activity 1: Students complete their games

Students wrap up design of their games. They **write** a rationale for their games that include:

- The name of their game
- The teaching goal of the game
- A brief paragraph on the science in the game, and the ways in which it connects to climate change
- A brief paragraph on the design decisions they made (e.g., We wanted to include a second level but it was too hard/not enough time/etc.)
- A brief response to the question: "If I had been expert in Scratch I would have included (what I wish my game had been)…"

Activity 2: Students prepare for an audience

To present their game, student pairs should prepare answers to the following questions, and decide between themselves who will present what:

1. What is the name of your game?
2. What specific climate change science will the player learn while playing the game?
3. How will the player learn the science?

4. What is the best feature of your game?
5. If you had been an expert in Scratch, what do you wish you could have included?

A worksheet for this can be found in the Handouts and Worksheets section.

Activity 3: Students showcase their games

Students give a brief presentation to introduce their games to the audience, play them to demonstrate, and then get to see others play and learn from their game.
Possible venues for showcasing (in addition to sharing online in the Scratch community):
- School science fair
- Graduation projects
- Parent night
- Class session with a younger grade.

Exit ticket

Optional: Each student can hand in the Game Presentation worksheet they used to prepare their game for final summative purposes (in addition to their game).

Handouts and Worksheets

Section 2: The Bathtub Model

Section 3: How to Give a Three-minute Talk

Section 4: What are good games?

Section 5: Experience Survey

Section 5: Create an account in Scratch

Section 5: The 10-block Challenge

Section 6: Requirements for Your Game

Section 6: Design Sketch Template

Section 7: Additional Scratch Tips

Section 7: Critique Form

Section 8: Rubric for Game Design

Section 8: Presentation of your game

SURVEY

STUDENT EXPERIENCE WITH SCRATCH

Name:______________________________

Date:__________________________

Have you had experience programming in Scratch?
Please check one of the following:

1. _____Huh? What is Scratch?

2. _____I've heard of it.

3. _____I used Scratch during Hour of Code in _________________ year.

4. _____I have created a project in Scratch.

5. _____I have a lot of experience in Scratch.

THE BATHTUB MODEL

Student name:

1. Explain what is represented by:
i) The bathtub

ii) The water

iii) The flow rate of water entering the bathtub

iv) The flow rate of water leaving the bathtub

v) The level of water in the bathtub

2. Why does the level of water in the bathtub continue to rise after the rate of emissions levels off?

3. How does the model falls short in representing the current imbalance in Earth's carbon cycle?

HOW TO GIVE A THREE-MINUTE TALK

You and a classmate are going to create a three-minute speech about all the topics in the learning library that are related to climate change. The goal is to share this task so that you get a quick overview without having to read all the topic overviews. This will help you and your partner to choose what topic your game will focus on. After that you will be able to explore your chosen topic in more depth as you design your game.

Here are some guidelines:

1. Find the Main Idea: Identify the main idea in the topic overview – your presentation should have *one* message. Next, identify 2-3 ideas or topics related to the main one.

2. Get Organized: Organize the main topic and supporting ideas into a structure:

- Introduction – the main topic
- Body – 2-3 ideas related to the main topic
- Conclusion – repeat the main idea.

3. Get Ready to Present:

- Read it through at least 3 times, so you're reasonably familiar with what you want to say. Then you'll be able to look at your audience when you talk to them.
- Try to vary your tone of voice – monotones put people to sleep!
- Speak clearly, and avoid "ums" and "like" – these are fine in everyday speech but distracting to your audience.

WHAT ARE GOOD GAMES?

Student name:

Date:

1. What makes the game work? (What are the parts? How do they work together?)

2. Is the game easy? Hard? Why do you think so?

3. Does the game include a system? If so, what system?

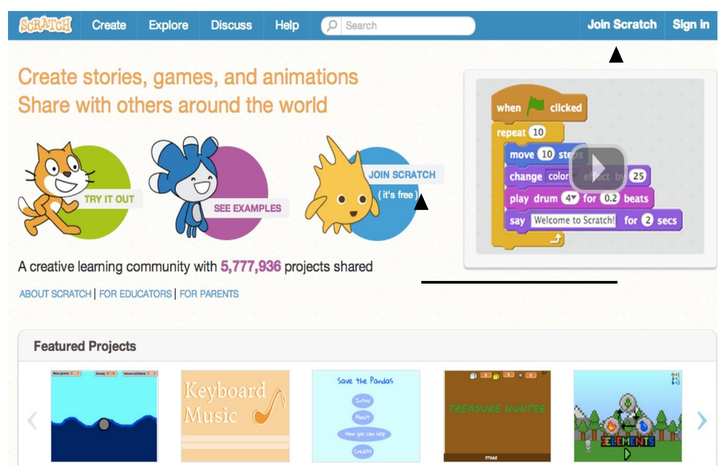
4. What can you learn from the game?

Create a SCRATCH ACCOUNT

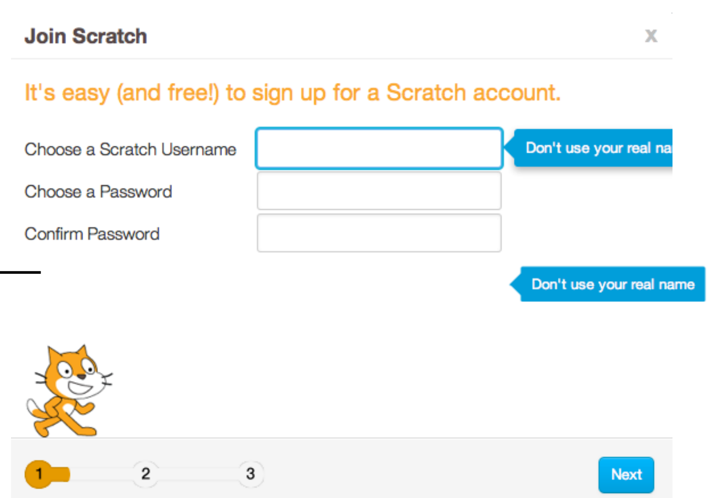
NEW TO SCRATCH? GET STARTED BY CREATING YOUR SCRATCH ACCOUNT!

You will need a Scratch account to create, save, and share your Scratch projects. The steps below will walk you through creating an account and setting up your profile.

- ☐ Open a web browser and navigate to the Scratch website: http://scratch.mit.edu
- ☐ On the homepage, click on "Join Scratch" at the top on the right or in the blue circle.



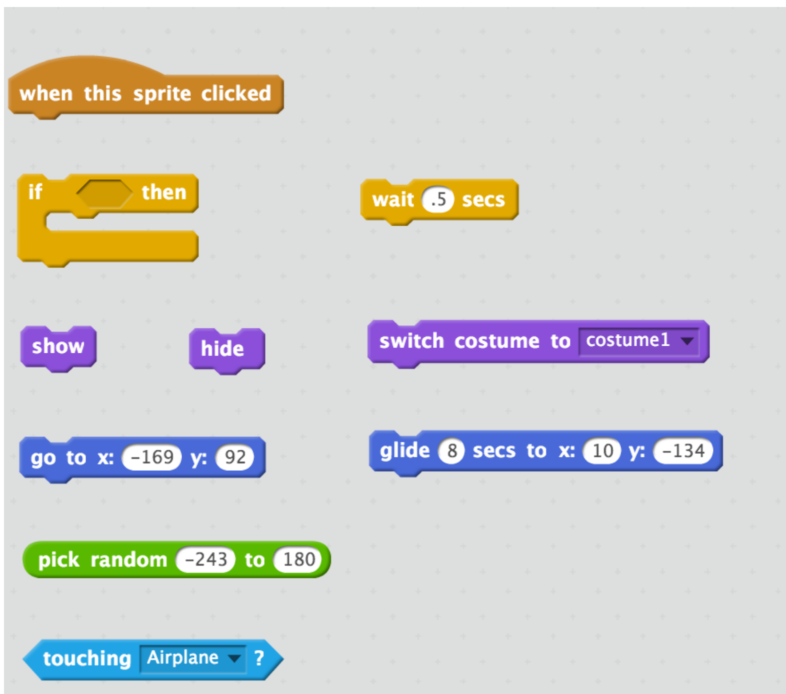
- ☐ Complete the three steps to sign up for your very own Scratch account!



THE 10 BLOCK CHALLENGE

Students with some Scratch experience do the 10-block challenge. They must create a project by remixing, using only these 10 blocks (shown in the screen below). They can use them once, twice, or multiple times, but must use each block at least once. Project the blocks in the classroom, or provide them with the illustration below.

Teacher Note: *A remix is a piece of media which has been altered from its original state by adding, removing, and/or changing pieces of the item. A song, piece of artwork, book, video, or photograph can all be remixes. The only characteristic of a remix is that it appropriates and changes other materials to create something new (Wikipedia). Remixing is a good strategy for programmers - students included. Some students may not think of it, or may consider it "cheating."*



REQUIREMENTS FOR YOUR GAME

The following requirements MUST be met:

1. The game must relate to some aspect of climate change
2. Designers must consider:
   -Reality (how can we accurately represent climate change)
   -Meaning (what will the player learn – the goal of the game)
   -Play (what will be the look and feel of game play on screen)
3. Have rules
4. Include a human connection.

DESIGN SKETCH TEMPLATE

1. REALITY: Sketching your climate change topic

Student names:

Topic What will be the overall climate change topic of your game? What specific topics do you want it to be about? How will *humans* act in your game?
System What parts of what system will you design in your game, and how do they interact with each other? Draw a systems diagram to show the system.

2. MEANING: Sketching the purpose of your game for the player

Purpose What do you want the player to learn or take away from the game?
Plan How will your game achieve this purpose?
Values What values or moral beliefs about climate change will be represented in your game? How?

PLAY: Sketching the genre of your game

Overall Goal What will the player do in your game (*e.g. find the flag, or save planet earth*)?
Gameplay What will be the rules of your game? What actions will players perform?
Game world What will be the story in your game? How will your game space look and feel on the screen? (*e.g. colors, sound, realistic or not?)*

ADDITIONAL SCRATCH TIPS

Backpack
Use the Scratch backpack to copy scripts or sprites from other people's projects and use them in your own. To put a sprite, script, sound, or backdrop into your backpack, you must drag them into the backpack area.
See the scratch wiki for more information: https://wiki.scratch.mit.edu/wiki/Backpack

Debugging
1. If you program the whole project and then test it, it is harder to debug and correct. Rather, use an iterative and incremental development approach:

- Divide the project by breaking it down to small independent goals
- Code one of the goals
- Test this goal
- Debug this goal

Once the goal is running as expected, go to the next one until all the goals have been achieved.

2. Some people find it useful to annotate the code with comments as they go, to know what each step is supposed to do.

3. Insert the "say" block in a script – this will help you to see what happens step by step, and to check values.

Getting information
1. Click on blue "i" on a sprite in the Sprites pane on the screen to get information about the sprite. The info pane also includes a few tools for manipulating the sprite.

- A sprite's name can be changed by clicking the box that contains the sprite's name.
- The selected sprite can be manipulated by pressing one of the following buttons:
    - Can Drag In Player — Checking this will make the sprite draggable outside the editor.
    - Circular Arrow — Enables full rotation in a sprite.
    - Linear Arrow — Enables the sprite to only face left or right.
    - "-" — Disables Rotation. The sprite will remain facing 90 degrees no matter what.
    - Pressing and dragging the blue line located over the sprite will rotate the sprite.
    - Double clicking the sprite will return it to facing 90 degrees.
    - The sprite's X and Y coordinates as well as direction are listed.
    - If the pen is down, the color is shown.

2. You can get information on any block of Scratch code by right-clicking on the block.

Importing backgrounds
When importing backgrounds, the pixel density of the backdrop is 360x480 pixels. To fit this exactly, the pictures need to be made this size before being imported to Scratch.

Saving
Scratch saves automatically but save often anyway!

CRITIQUE FORM

Feedback from: ____

Feedback for: ____

	What works well or you really like?	What is confusing?	What could be improved?
PLAY: How engaging is the project? How did you feel as you interacted with it?			
MEANING: Is the learning goal obvious to you?			
REALITY: Is the climate change topic clear?			

Names:______________________________________Game: ________________________________

Final Project Rubric

Scoring Category	Criteria for an excellent project
Reality (10 pts)	• Accurately represents the climate system and/or human interactions within it. • Demonstrates complexities within climate change systems. (ex. Interactions, inflow/outflow rates, feedback loops, human decisions and dilemmas). • Player action affects the climate system in realistic (relative to Scratch experience) ways. • Includes a connection between human actions and climate change.
Meaning (10 pts)	• Goal of the game is clearly explained. • Peer audience can identify parts of the climate system from your game. • Game achieves its goal.
Play (10 pts)	• It is clear how to play the game and what the end goal is for the player. • Game is engaging, players want to play through to the end. • Creative ideas about how to improve the game are presented.
Using Scratch (10 pts)	• Graphics creatively support game play. • Code uses variables that interact with one another. • Game is debugged and runs as intended considering Scratch experience and time provided.
Effort (5 pts)	• Consistently on task during project work time. • Demonstrates problem-solving skills and advocates appropriately when help is needed.
Teamwork (5 pts)	• Driver/navigator roles and responsibilities equally shared. • Communicates effectively with partner. • Maintains positive attitude when obstacles encountered. • Supportive and encouraging as a partner.

Total: 50 pts.

Comments:

GAME PRESENTATION

Names:____

1. What is the name of your game?

2. What specific climate change science will the player learn while playing the game?

3. How will the player learn the science?

4. What is the best feature of your game?

5. If you had been experts in Scratch, what else would you have liked to include?