



# **HOUR OF CODE 2021 (TimeCraft)**

---

## **EDUCATOR GUIDE**

This Educator Guide includes:

- Getting Started
- Hour of Code 2021 Theme Overview
- Learning Goals & Lesson Objectives
- Suggested Teacher Experience
- Learning Experiences for Students
- Lesson Activities
- Educational Standards
- Extension Activities
- Visual Glossary

**[EDUCATION.MINECRAFT.NET](https://education.minecraft.net)**

# TABLE OF CONTENTS

INTRODUCTION	3
HOUR OF CODE 2021: THEME OVERVIEW	4
LEARNING GOALS	5
SUGGESTED TEACHER EXPERIENCE	6
LEARNING EXPERIENCES FOR STUDENTS	7
LESSON ACTIVITIES	9
EDUCATIONAL STANDARDS	20
EXTENSION ACTIVITIES	22
KAHOOT!	23
VISUAL GLOSSARY	24

## Introduction

***Are you ready for an adventure through time?*** Welcome to the Hour of Code 2021 (TimeCraft)!

This Educator Guide is designed to provide you with all the information you need to help your students have an enjoyable and successful Hour of Code.

Hour of Code 2021 (TimeCraft) has been designed as a fun and creative tutorial introduction to computer science (CS). This one-hour experience will provide your students the basics of coding (Block or Python) and demonstrate all the ways they can participate in CS.

This year's Hour of Code 2021 also offers additional opportunities to replay and learn more coding skills and connect to other areas. We have added extension lessons and Kahoot! activities to extend learning and fun for all learners!

**ALL students can participate in the Hour of Code 2021!** Let's go!

## How to Get Started

- Install Minecraft: Education Edition by visiting [aka.ms/HourofCode2021](https://aka.ms/HourofCode2021)
- Use this Educator Guide!
- Watch [the introductory video](#) to this year's Hour of Code theme.
- Watch [the walkthrough video](#) to get a better understanding of the lesson and how to navigate through the Minecraft world.
- Download a copy of the [coding solutions](#).
- Do you have additional questions? Check out the [FAQ](#).

# Hour of Code 2021: TimeCraft—Theme Overview

**Computer science is everywhere. Computer science is for everyone.**

Take a look around you... did you know that computer science can help you create songs in music class? Did you know that computer science can help you research information for your science experiment? Did you know that computer science can help your school keep track of daily attendance?

Computer science can be used in many different ways. Computer science concepts have a wide range of applications in our everyday lives. It has changed the way we live, work, and even, play! From automation in factories to online databases in schools, computer science can be found in every job, career path, and industry. Computer science is a part of every passion, interest, hobby, and career.

We also use computer science skills to help us solve problems, advance research, create artwork, apply mathematics, design buildings, stay healthy, and so much more.

Join us as we explore computer science—everywhere and for everyone!

## **Hour of Code 2021: TimeCraft**

As a computer scientist for the Institute of Major Time Errors, it's your job to correct the mysterious Time Splits appearing in history and to find who (or what!) is causing them.

Will you help fix the Time Splits and save history using your coding superpowers? Will you find out the truth about who, or what, is causing these crazy changes in history?

In your TimeCraft mission, you need to:

- Travel back to exciting moments in world history
- Code your Time Agent to fix the Time Splits
- Use the clues to identify the Culprit (who or what is causing the Time Splits)

## Learning Goals



Understand the importance and benefits of computer science in all aspects of our lives



Apply concepts of algorithmic thinking and problem decomposition in Hour of Code and other content areas



Iterate on coding solutions to create a program to successfully complete a task



Build awareness around CS career connections

## Suggested Teacher Experience



### 1. HOUR OF CODE: GAME PLAY

This is when students will actively engage in game play. You will utilize the [lesson activities](#) and corresponding presentations to help successfully facilitate HOC.

### 2. EXTENSION ACTIVITIES

The [ten extension activities](#) will create meaningful content connections for integrated computer science experiences. Extending the game play experience has the ability to:

- Provide contextual understanding of the historic events students experience within the game
- Allow for seamless integration of content areas (literacy, mathematics, science, art, music, history, digital fluency)
- Connect careers and interests and the relevance of computer science

### 3. KAHOOT!

Use this [interactive and engaging resource](#) to extend your students' thinking by testing out your existing knowledge in various content areas such as history, music, visual arts, science, and more!

### 4. BEYOND THE HOUR OF CODE: LESSON SUGGESTIONS

Sustaining your students' interests is easy with our [Beyond the Hour of Code](#) lesson suggestions. These suggested learning pathways encourage exploration and support student learning beyond the initial HOC experience.

This also affords you, as the teacher, with easy-to-implement and high-quality content to support your existing curriculum.

## Learning Experiences for Students

In an effort to increase student access to the Hour of Code, there are three different learning experiences for students: in-class with a teacher facilitator, in-class as a self-guided experience, or in a remote (virtual) learning capacity. Each of the learning experiences have different levels of teacher support and modification for student success and participation in the Hour of Code.

### Teacher-Facilitating (Face to face)

***This type of learning experience is going to provide students with the highest level of teacher support.*** As the teacher, you will be providing explicit instruction and modelling before releasing students into independently working through the coding activities in the Hour of Code. You will utilize the [Hour of Code Presentation](#) to guide you and your students through beginning game play, coding activities, and lesson reflection.

### Student Self-Guided

***This type of learning experience is going to provide students with the least amount of teacher support.*** As the teacher, you will make sure your students are able to log into the Minecraft: Education Edition platform. You will provide students with the Hour of Code Presentation to assist them with their self-paced and self-guided experience for game play. The presentation will provide students with explicit directions on how to participate in the Hour of Code. You should also provide students with a copy of the Visual Glossary, which is located at the end of this Educator's Guide. It is recommended that you provide these materials to students in a digital medium of your choice or print off paper copies so they are readily available for student use.

### Remote Learning Experience

This type of learning experience can be facilitated synchronously or asynchronously.

***If you are having students participate in a synchronous (live) virtual session,*** you will facilitate the session just like the in-class experience (with teacher facilitation). As the teacher, you will be providing explicit instruction and modelling before releasing students into independently working through the coding activities in the Hour of Code. You will utilize the Hour of Code Presentation to guide you and your students through beginning game play, coding activities, and lesson

reflection. Make sure your virtual communication platform (like Microsoft Teams) is ready to share content and the sound is enabled.

***If you are having students participate in an asynchronous (i.e., not live) virtual session,*** you will facilitate the session just like the self-guided in-class experience. As the teacher, you will make sure your students have access to the Minecraft: Education Edition platform prior to lesson. You will also need to provide students with the Hour of Code Presentation to assist them with their self-paced and self-guided experience for game play. The presentation will provide students with explicit directions on how to participate in the Hour of Code. You should also provide students with a copy of the Visual Glossary, which is located at the end of this Educator's Guide. It is recommended that you provide these materials (presentation and glossary) to students in a digital medium of your choice (examples: Microsoft Teams channel, OneNote notebook, your learning management system, etc.).

If you plan to offer this lesson via remote learning, please consider these tips:

1. This lesson is not designed as a multiplayer experience. Each individual student should work in their own version of the world.
2. Divide students into pairs or small groups via breakout rooms so they can help one another troubleshoot as they solve the coding activities.
3. Given that students are likely to have varying levels of familiarity with Minecraft and coding, it may help to assign student group leaders to help peers who may get stuck.

For more information on remote learning in Minecraft: Education Edition, please refer to <https://aka.ms/remote-learning-kit>



## Lesson Activities


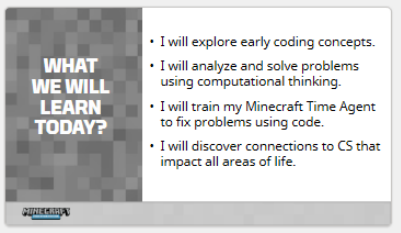
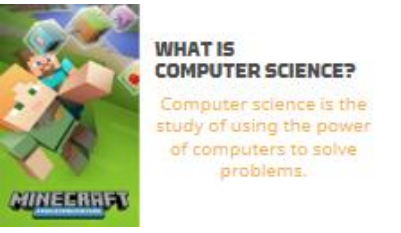
This next section will provide you with an overview of exactly how to facilitate the lesson activities for the Hour of Code.



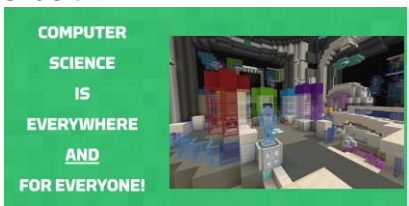

There will be three different parts:

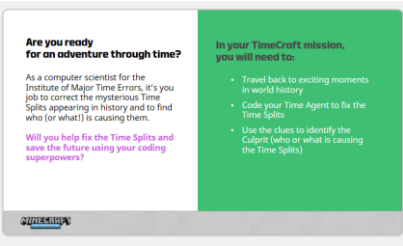


- Direct Instruction (I Do)
- Guided Instruction (We Do)
- Independent Work (You Do)

This structure is an intentional lesson design following the gradual release model to ensure student success. Both the Direct Instruction and Guided Instruction parts will utilize the Hour of Code Presentation. The annotated notes found in this document will directly correlate with the slides with the [HOC presentation](#). You should use them together.



### Direct Instruction (I Do...) (teacher led)



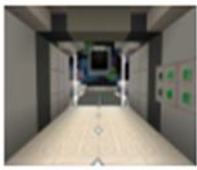



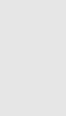



<p>Slide 1</p> 	<p>At the time, you should have this presentation on display. The next several slides are intended to set the stage for the HOC game play experience. Students are not logged into the platform yet. However, you should provide the visual glossary (provided at the end of this guide) for reference.</p>
<p>Slide 2</p> 	<p>Review the “I will...” statements with students. (Read slide)</p>
<p>Slide 3</p> 	<p>Ask the question to students. Allow for students’ responses. (Answers will vary) Some possible responses may include: coding, programming, creating apps, creating software, learning how hardware and software work together, etc.</p>

<p>Slide 4</p> 	<p>Ask the question to students. Allow for students' responses. (Answers will vary) Some possible responses may include: school attendance (if it is taken online), learning management system, school databases, library resources, technology spaces (or facilities), etc.</p>
<p>Slide 5</p> 	<p>Ask the question to students. Allow for students' responses. (Answers will vary) Some possible responses may include: it helps people to communicate (i.e., video chat, conference calls, emails, online messaging, etc.), to track work production, marketing, client outreach, etc.</p>
<p>Slide 6</p> 	<p><i>"Computer science is everywhere! Computer science is for everyone. Take a look around you... did you know that computer science can help you create songs in music class? Did you know that computer science can help you research information for your science experiment? Did you know that computer science can help your school keep track of daily attendance? Computer science can be used in many different ways. Computer science concepts have a wide range of applications in our everyday lives. It has changed the way we live, work, and even, play! From automation in factories to online databases in schools, computer science can be found in every job, career path, and industry. We also use computer science skills to help us solve problems, advance research, create artwork, apply mathematics, design buildings, stay healthy, and so much more."</i></p>
<p>Slide 7</p> 	<p>This is the 2-minute trailer video. Advance the slide to play the trailer.</p>

<p>Slide 8</p> 	<p>(Read the slide to students)</p>
<p>Slide 9</p> 	<p>There are some very important words for us to understand in order to follow the storyline in Hour of Code. Let's look at these three words together. (Read the vocabulary words and definitions to students)</p>
<p>Slide 10</p> 	<p>(Read the slide to students)</p> <p>Then help all students to log into the Minecraft: Education Edition platform. After logging in, they should launch the HOC 2021: TimeCraft world. Explain to students that they should NOT start playing- just simply launch the platform.</p> <p><b>NOTE:</b> This is a single-player game. Although they will all be seeing the same world, each person will be playing their own adventure.</p>


## Guided Instruction (We Do...) (Class activity)

<p>Slide 11</p> <p><b>WELCOME TO TIMECRAFT</b></p>  <p>This is your spawn point, the location where you begin game play.</p>	<p><i>"This is your spawn point—the point where you begin game play. All of you should see this exact image. Do not push anything yet. We are going to watch a short clip on how to move and use the basic controls in Minecraft before we begin."</i></p>
<p>Slide 12</p> 	<p>Depending on the type of your devices, you should select the appropriate one: keyboard or touch controls.</p> <p>Play the video for students to get a brief summary of how to move and execute basic controls in Minecraft. This will help students understand how to navigate.</p>

<p>Slide 13</p> <div data-bbox="203 226 652 478"> <p><b>LET'S PRACTICE MOVING...</b></p>  <p>Use your controls to navigate through these obstacles and make it back to this green button shown here.</p> </div>	<p>Instruct students to push the button on the gold block to open the doorway. Once they press the button, this is the image they will see in front of them. It's now time to start moving.</p>
<p>The next several slides are intended to be played through and modelled for students. The teacher should be using these slides to provide step-by-step directions for students. Students should follow each step after it has been demonstrated by the teacher.</p>	
<p>Slide 14</p> <div data-bbox="267 682 717 934"> <p><b>PUSH THE BUTTON</b></p>  <p>Buttons are going to be a VERY important part of game play.</p> <p>To push a button, walk up to the button and right-click on the button (if using a keyboard) or simply just touch the button (if using touch controls).</p> </div>	<p>Slide 15</p> <div data-bbox="885 682 1334 934"> <p><b>WALK THROUGH THE HALLWAY</b></p>  <p>This is the view you will see after you push the button on the unlock door.</p> <p>Follow the trail through hallway to the Control Center - you are now in the Institute for Major Time Errors. You will need this room throughout game play.</p> </div>
<p>Slide 16</p> <div data-bbox="267 987 717 1239"> <p><b>MEET TARRA</b></p>  <p>Meet TARRA!</p> <p>She is an artificial intelligence (AI) robot. She is a computer robot that is able to think and complete tasks like a human. She will be your guide throughout TimeCraft.</p> </div>	<p>Slide 17</p> <div data-bbox="885 987 1334 1239"> <p><b>IMMERSIVE READER</b></p>  <p>Immersive Reader can be opened inside Minecraft to read signs and NPC dialogue! It is incredibly helpful!</p> <p>It's great for extra help for reading, including translating the text and reading it aloud.</p> <p>How to use Immersive Reader in Minecraft: Education Edition <a href="https://aka.ms/ImmersiveReaderFE">https://aka.ms/ImmersiveReaderFE</a></p> </div>
<p>Slide 18</p> <div data-bbox="267 1291 717 1543"> <p><b>Code is a set of instructions.</b></p> <p>These instructions can be written in different ways:</p> <div data-bbox="332 1344 609 1491"> <p><b>Block-based code</b></p>  </div> <div data-bbox="625 1344 706 1491"> <p><b>Text-based code</b></p>  </div> <p>You will need to choose between Blocks or Python. (Beginners are recommended to start with Blocks)</p> </div>	<p>Slide 19</p> <div data-bbox="885 1291 1334 1543"> <p><b>TARRA AND NPCs</b></p>  <p>Throughout TimeCraft, you will see TARRA and other NPCs (non-player characters).</p> <p>TARRA will tell you important information to help you during game play. You will see pop-up screens from TARRA - these will help guide you through the game.</p> </div>
<p>Slide 20</p> <div data-bbox="267 1606 717 1858"> <p><b>PICK A TIME AGENT</b></p>  <p>It is now time to pick a Time Agent.</p> <p>There are five different colors. You can choose whichever color you like!</p> </div>	<p>Slide 21</p> <div data-bbox="885 1606 1334 1858"> <p><b>HOW TO PICK A TIME AGENT</b></p>  <p>Push the button to choose Time Agent.</p> </div>

Slide 22


### USING YOUR TIME AGENT



The Time Agents will disappear. You will need to press the "C" button to launch your agent (or tap the agent icon for touch).

Slide 23

### CODE BUILDER (BLOCKS)




Once you push the "C" button, you will launch Code Builder.

Code Builder is the coding palette you will use to program your Time Agent to help you solve problems throughout TimeCraft.

Press the green arrow to return to game play.

Slide 24

### CODE BUILDER (PYTHON)




Once you push the "C" button, you will launch Code Builder.

Code Builder is the coding palette you will use to program your Time Agent to help you solve problems throughout TimeCraft.

Close this window to return to game play.

Slide 25


### USING YOUR TIME AGENT



Your Time Agent is ready!

Slide 26

### MESSAGE FROM TARRA



Go to the Test Center.

Slide 27


### GO TO THE TESTING CENTER



Follow the trail to the Testing Center.

Slide 28

### MESSAGE FROM TARRA



That was fast! Look! I have added the TALK device in your inventory. Use the TALK to call your Time Agent or when you need me. Right click with the device in your hand to activate it.

Slide 29

### USE YOUR TALK DEVICE




Your TALK device is located in your hotbar.

Select the TALK device in the hotbar. Once you select the TALK device, you will see a new button on screen: **Use TALK device**

Press this button.

Slide 30


### MESSAGE FROM TARRA



Now I'm only a click away! Press "Start Activity" to begin!

Slide 31

### MESSAGE FROM TARRA



Code to move your Time Agent forward 3 blocks to test it.



Slide 32

**Coding concept:**  
Sequencing


What it means

The Agent will move in the order you sequenced. In a sequence structure, an action or event leads to the next ordered action.

Let's think about it

Would your last program work if you put agent analyze before agent move? Why or why not?

Tip: Try planning on paper!



Slide 33

**TEST THE AGENT**



Press "C" to open Code Builder.

Slide 34

**USING CODE BUILDER (BLOCKS)**



Step 1: Read the coding task.

Step 2: Use the MakeCode blocks from your toolbox. You will drag and drop them into the coding canvas.

Step 3: Press the green start arrow to test your code.

Slide 35

**USING CODE BUILDER (BLOCKS)**



If you need additional help to solve a coding challenge, select the **hint** button, a lightbulb with a question mark. These hints are very helpful!

Slide 36

**USING CODE BUILDER (PYTHON)**




Step 1: Read the coding task.

Step 2: Type in your Python code under Activity.

Step 3: Press the "run" button on the bottom of the screen.

Slide 37

**MESSAGE FROM TARRA**



Well done! Your Time Agent has followed your code.


Slide 38

**GO TO THE TIMELINE COMPUTER**



Slide 39

**MESSAGE FROM TARRA**



Let's try to solve this problem. Press the button to select it on the Timeline Computer.

Slide 40


**SELECT THE TIME SPLIT**



Push the button to choose the Time Split.

Slide 41

**MESSAGE FROM TARRA**



You have selected a Time Split. Now, you should move to the Time Pod to begin your mission.

Slide 42

**GO TO THE TIME POD**




The Time Pod is a machine that allows you to time travel!




Slide 43

**MESSAGE FROM TARRA**



Move toward the Time Pod and press the button ahead to open it and receive your mission brief!



Slide 44

**TIME JUMPS**



You will have to solve 3 Time Splits. Every time you enter the Time Pod, the number shown on the side will tell you how many time jumps you have to solve.

Push the button to open the Time Pod.



Slide 45

**ENTER THE TIME POD**



Push the button in the Time Pod.



Slide 46

**WELCOME TO BIG BAND JAZZ!**




This is your spawn point in this world. You will see this exact image when you start game play.




Slide 47

**MESSAGE FROM TARRA**




This is bad. If the musician cannot find his trumpet, jazz music may never exist. Let's see if we can help him out?

I'm detecting something belonging to the Jazz Musician behind a wall. Start the mission by using your TALK device to call your Time Agent.




Slide 48

**JAZZ MUSICIAN NPC**



Booted off another stage! Ever since I lost my trumpet, nothing has been the same! The kazoo just doesn't have the same zing!



Slide 49

**BIG BAND JAZZ**



Once you have the TALK device in your hand, you will see a new gray box saying "Use TALK device".

Press the button to see the next pop-up screen.




Slide 50

**BIG BAND JAZZ**




Click on "Start Activity" to begin your mission!




Slide 51

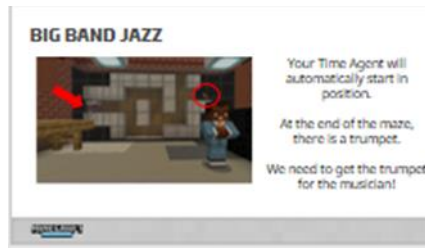
**MESSAGE FROM TARRA**



Excellent! You've revealed the location of the Jazz Musician's trumpet. Use code to move the Time Agent through the maze and collect the trumpet.



Slide 52



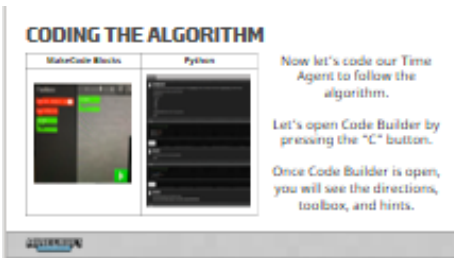
Slide 53



On Slide 53, you are going to model your own think aloud for how the Time Agent would move through the maze.

- For the first command, the Time Agent must move forward.
- How many times should the Time Agent move forward?
- Then what should happen next?
- Is there more than one way to complete this task?
- What are the different ways to complete this task?
- What is the complete algorithm (using the commands)?

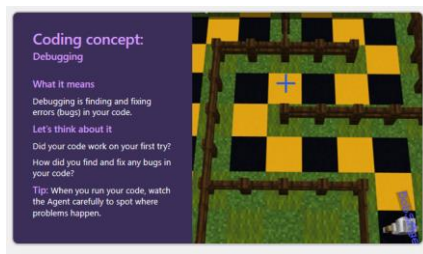
Slide 54



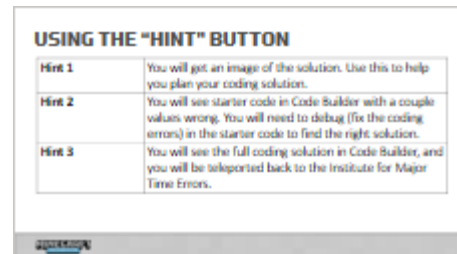
Slide 55



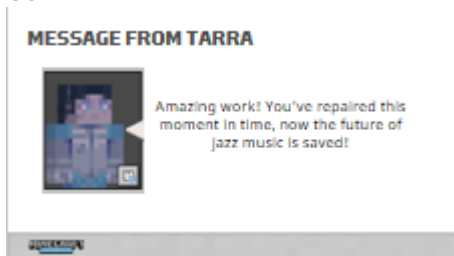
Slide 56



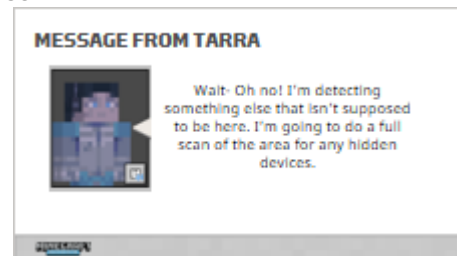
Slide 57



Slide 58



Slide 59





Slide 60


### SECRET REVEALED



After TARRA scans the room, the secret button has been revealed!

Slide 61

### MESSAGE FROM TARRA



My scan has revealed a hidden button somewhere in this area!

Search the area from top to bottom and then press the button to gather more information!

Slide 62

### SECRET BUTTON



There is a secret button hidden in this Time Split. Follow the trail to the secret button.

Slide 63

### SECRET BUTTON




This is the location of the secret button.

Once you click on this button, it will open the white doors on the left.

Slide 64

### MESSAGE FROM TARRA




You've found the hidden button!

It appears a secret room has been revealed in this area. It must be the Culprit's hideout!

Go to the room and search it for any clues!

Slide 65

### RETURN TO THE CONTROL CENTER




Let's starting narrowing down which Time Agent could be the Culprit.

Based on the evidence you've gathered, which one do you think is causing all of the Time Splits?

Slide 66

### VOTING



After each Time Split, you will have the opportunity to vote for who you think the Time Culprit is.

Slide 67

### SUCCESS!

BEFORE	AFTER
	

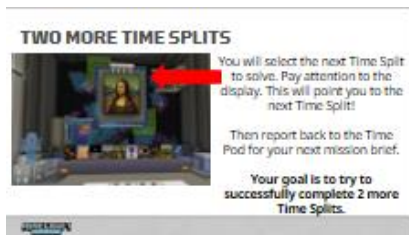
Slide 68

### TROPHY AREA

Go explore the Control Center to find the Trophy Area. Each time you successfully solve a time split, a trophy will appear!



## Independent Work (You Do...)



At this point, you are going to allow students to select their next Time Splits on the mainframe and continue through game play.

Students have the possibility of completing two additional Time Splits during the Hour of Code.

### Overview of the Time Splits

<b>Time Split 1</b> Big Band Jazz (This is the Guided Challenge)	The great jazz musician has lost his beloved trumpet and has replaced it with a kazoo! Code your Time Agent to get through the maze and retrieve the musician's trumpet to save jazz.
<b>Time Split 2</b> The Great Pyramids of Giza	The Great Pyramids are now cubes! Code your Time Agent to help the designer create a secure structure that will last for thousands of years... the pyramid.
<b>Time Split 3</b> Moon Mission	The astronauts need your help—use your Time Agent to complete a faith maze and deliver the calculations to help the astronauts land on the Moon!
<b>Time Split 4</b> The Great Wall of China	The Great Wall isn't actually great yet... In fact, it's very short! The hungry panda bears keep eating the bamboo scaffolding, which now means the workers cannot build the great and tall wall they wanted. Code a bamboo garden to divert the hungry pandas!
<b>Time Split 5</b> The Mona Lisa	Mona Lisa is no longer smiling—in fact, she is very upset. The Time Culprit went back and trampled Mona Lisa's garden, causing her famous smile to be a frown. Code your Time Agent to cheer her up by planting more flowers.
<b>Time Split 6</b> First Flights	The Culprit has filled the airplane runway with holes. Instead of having airplanes, we only have hot air balloons. Code your Time Agent to fix the runway so the inventors can make their first flight!

<b>Time Split 7</b> First Computer Scientist	The first computer science program played a song, but the Culprit has ruined the code. Use your Time Agent to fix the code to make the music play!
<b>Time Split 8</b> Human's Best Friend	Human's best friend is no longer the dog, but a bear! Use your Time Agent to befriend a pup and lead a pup back to the humans to become best friends again.
<b>Time Split 9</b> Palaeontology Puzzle	Oh no! The Culprit went back in time and stole some fossils causing the brachiosaurus to have a short neck. Use your Time Agent to help replace the stolen fossils.
<b>Time Split 10</b> Elements of Discovery	The culprit snuck into the science lab and hid some of the elements. Use your Time Agent to find the hidden elements so the scientist can make the great discovery!

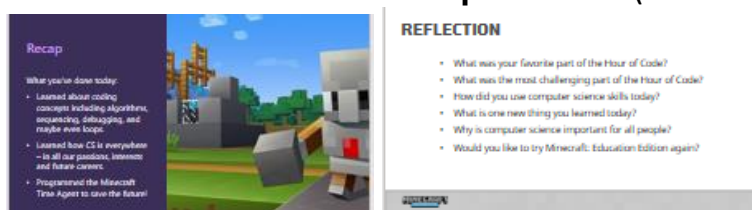
## Completing the Hour of Code

After students have completed the two additional time splits, they will experience the completion of the Hour of Code. (slides 71-72)

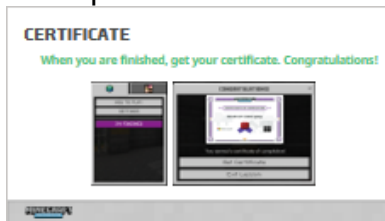


## Reflection & Celebration

After game play has finished, gather students back together to recap their learning and to discuss the **reflection questions**. (slides 73-74)



Then provide students with their **certificate of completion**.



## Educational Standards

<b>Standards</b>		
<b>United States</b>	<b>Australia</b>	<b>UK - England</b>
I can model daily processes by creating and following algorithms to complete tasks  <b>1A-AP-08</b>	I can follow, describe and represent a sequence of steps and decisions (algorithms) needed to solve simple problems  <b>ACTDIP004</b>	I can create and debug simple programs  <b>Key Stage 1</b>
I can develop programs with sequences and simple loops, to express ideas or address a problem  <b>1A-AP-10</b>	I can define simple problems, and describe and follow a sequence of steps and decisions (algorithms) needed to solve them  <b>ACTDIP010</b>	I can understand what algorithms are, how they are implemented as programs on digital devices, and that programs execute by following precise and unambiguous instructions  <b>Key Stage 1</b>
Debug (identify and fix) errors in an algorithm or program that includes sequences and simple loops.  <b>1A-AP-14</b>	Design, modify and follow simple algorithms involving sequences of steps, branching, and iteration (repetition)  <b>ACTDIP019</b>	I can use logical reasoning to predict the behaviour of simple programs  <b>Key Stage 1</b>
I can compare how people live and work before and after the implementation or adoption of new computing technology  <b>1A-IC-16</b>	Design algorithms represented diagrammatically and in English, and trace algorithms to predict output for a given input and to identify errors  <b>ACTDIP029</b>	I can use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs  <b>Key Stage 2</b>
I can create programs that include sequences, events, loops, and conditionals.  <b>1B-AP-10</b>	I can define simple problems and describe and follow a sequence of steps and decisions (algorithms) needed to solve them.  <b>ACTDIP01</b>	I can use sequence, selection, and repetition in programs; work with variables and various forms of input and output  <b>Key Stage 2</b>
I can compare and refine multiple algorithms for the same task and determine which is the most appropriate  <b>1B-AP-08</b>	I can implement simple digital solutions as visual programs with algorithms involving branching (decisions) and user input  <b>ACTDIP011</b>	I can design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts  <b>Key Stage 2</b>
Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended.  <b>1B-AP-15</b>	I can implement digital solutions as simple visual programs involving branching, iteration (repetition), and user input  <b>ACTDIP020</b>	I can understand several key algorithms that reflect computational thinking; use logical reasoning to compare the utility of alternative algorithms for the same problem  <b>Key Stage 3</b>

<p>I can discuss computing technologies that have changed the world, and express how those technologies, and are influenced by, cultural practices</p> <p><b>1B-IC-20</b></p>	<p>I can implement and modify programs with user interfaces involving branching, iteration and functions in a general-purpose programming language</p> <p><b>ACTDIP030</b></p>	<p>I can use 2 or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures; design and develop modular programs that use procedures or functions</p> <p><b>Key Stage 3</b></p>
<p>I can use flowcharts and/or pseudocode to address complex problems as algorithms</p> <p><b>2-AP-10</b></p>	<p>I can collect, access and present different types of data using simple software to create information and solve problems</p> <p><b>ACTDIP009</b></p>	
<p>I can design and iteratively develop programs that combine control structures, including nested loops and compound conditionals</p> <p><b>2-AP-12</b></p>		
<p>I can decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs</p> <p><b>2-AP-13</b></p>		

## Extension Activities

The ten [extension activities](#) are lesson plans to create meaningful content connections for integrated computer science experiences. Extending the game play experience has the ability to:

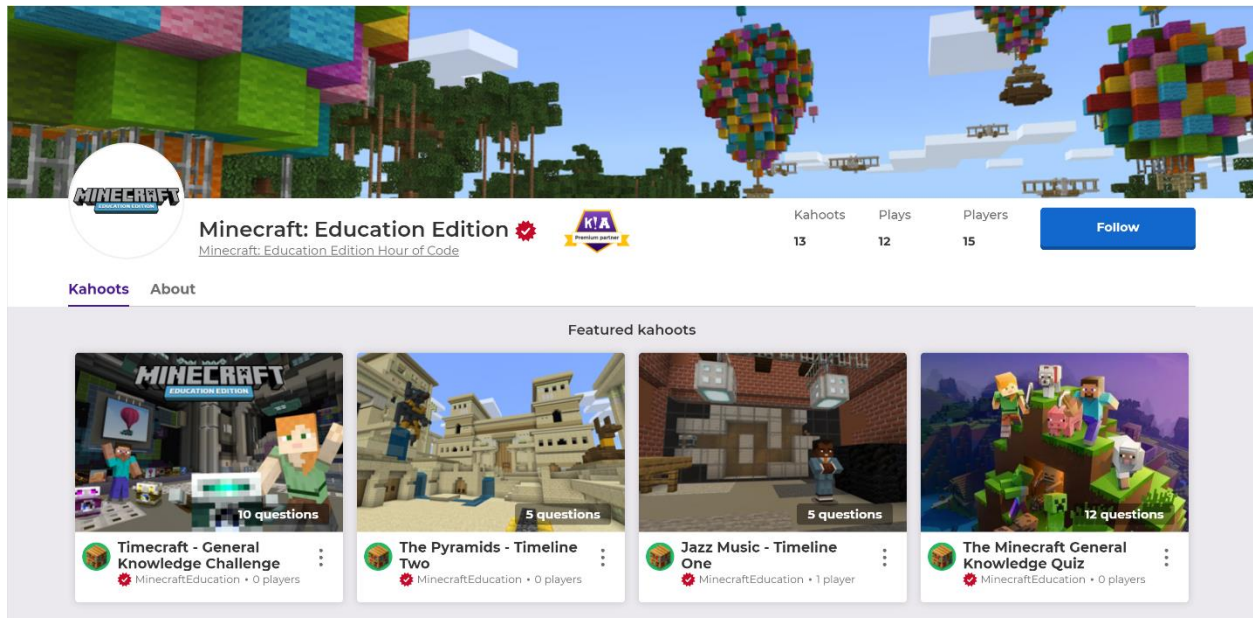
- Provide contextual understanding of the historic events students experience within the game
- Allow for seamless integration of content areas (literacy, mathematics, science, art, music, history, digital fluency)
- Connect careers and interests and the relevance of computer science

Each of the extension activities include a full lesson plan with correlating content standards addressed within the lesson.

### ACTIVITY OVERVIEW

<b>Extension Activity #1</b> Big Band Jazz	Students will explore the life of Louis Armstrong and his lasting influence on jazz music. (Music)
<b>Extension Activity #2</b> Pyramids of Giza	Students will explore the geometric shapes used in the construction of the pyramids. (Mathematics)
<b>Extension Activity #3</b> Moon Mission	Students will learn about Katherine Johnson and her remarkable accomplishments in the fields of mathematics, physics, aeronautical research, and computer science. (Mathematics)
<b>Extension Activity #4</b> The Great Wall of China	Students will learn about the history of the Great Wall of China and the construction materials used to build the Great Wall. (Science)
<b>Extension Activity #5</b> The Mona Lisa	Students will learn about Leonardo da Vinci's contributions to the art world and learn about his most influential work, "The Mona Lisa". (Visual Arts)
<b>Extension Activity #6</b> First Flights	Students will learn about The Wright Brothers' contributions to the world of aircraft and learn about engineering design principles that enabled the first flight to occur on Kitty Hawk in 1903. (Science)
<b>Extension Activity #7</b> First Computer Scientist	Students will learn about Ada Lovelace and her profound influence as the world's first computer programmer. (Computer Science/Digital Technologies)
<b>Extension Activity #8</b> Human's Best Friend	Students will learn the history behind the domestication of animals over time and how domestication has allowed us to own and care for pets. (Language Arts/Literacy)
<b>Extension Activity #9</b> Palaeontology Puzzle	Students will learn about the job of a paleontologists and how they study fossils to understand more about the dinosaurs (and other living organisms) of the past. (Science)
<b>Extension Activity #10</b> Elements of Discovery	Students will learn the importance of diversity in science and how Marie Curie helped to defy the stereotypes for women scientists. (Media Arts)

# Kahoot!



Extend your students' thinking by testing out their existing knowledge in the following content areas:

Time Split	Content Area	<a href="https://aka.ms/HOC2021Kahoot">aka.ms/HOC2021Kahoot</a>
Big Band Jazz	Music	
The Great Pyramids of Giza	Ancient History	
Moon Mission	Science	
The Great Wall of China	History	
The Mona Lisa	Visual Arts	
First Flights	Engineering	
First Computer Scientist	Computer Science	
Human's Best Friend	Life Science	
Palaeontology Puzzle	History + Science	
Elements of Discovery	Chemistry	



# MINECRAFT VISUAL GLOSSARY

## Chalkboards

special blocks that allow you to write and display text in M:EE worlds



## Chest

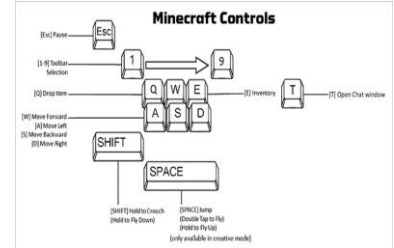
stores valuable items (outside of your inventory)



## Controls

(keyboard)

keyboard buttons that help you move around and complete tasks



## Controls

(touch)

the touch pad that helps you move around and complete tasks



## Creative Mode

one of the main game modes in Minecraft: gives the ability to fly and endless play



## Ephemeria One

the TimeCraft spaceship



## Hotbar

selection bar that appears on the bottom of the screen



## Inventory

press "E" to get this pop-up menu that you can use to find items



## Minecraft: Education Edition

a game-based learning platform



## NPC

non-player character



## Search Feature

allows you to search for specific items in the inventory by typing in the keyword



## Spawn Point

the location where a player begins game play





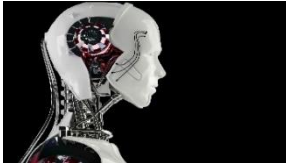
<p><b>TALK</b>  <b>(Time Agent Link Radio)</b>  a radio device that brings your Time Agent to you</p> 	<p><b>TARRA</b>  A.I. robot, who is your guide</p> 	<p><b>Time Agent</b>  your own personal robot</p> 
<p><b>Time Culprit</b>  the Time Agent causing all the trouble</p> 	<p><b>Time Pod</b>  a machine that allows you to time travel</p> 	<p><b>Time Split</b>  a break in time that changes the future</p> 

# COMPUTER SCIENCE GLOSSARY

## A.I.

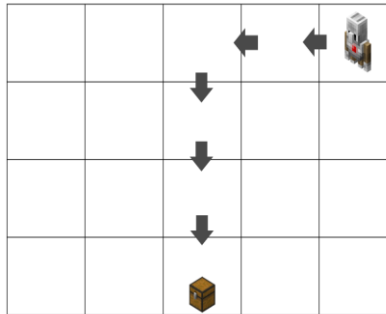
### (Artificial Intelligence)

A computer or machine that is able to perform tasks that normally a human does



## Algorithm

a sequence of defined steps



## Code (Coding)

the method of giving a computer instructions to perform a specific task



## Code Builder

coding palette used to create programs



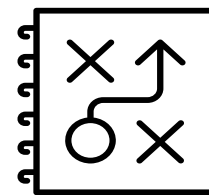
## Debug

the process of finding (and correcting) errors in a computer program



## Decompose

breaking down a problem or task into smaller, more manageable parts



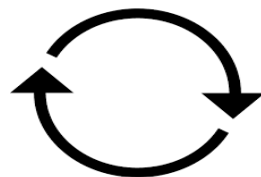
## Integer

a whole number; a number that is not a fraction

```
agent.give("dirt",64,1)
for i in range(2)
    agent.move("forward")
    agent.place(1, "down")
```

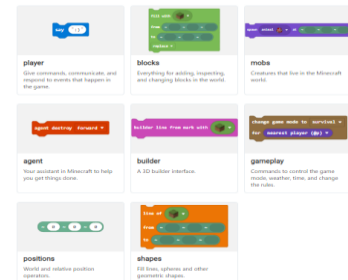
## Loop

sequence that repeats a portion of code a set number of times until the desired task is complete



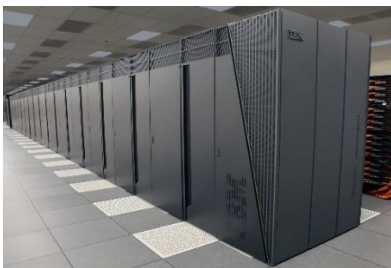
## MakeCode Blocks

blocks used to create code



## Mainframe

a large, highspeed computer



## Python

a computer programming language that uses a mixture of recognizable words and characters

```
def on_chat():
    for i in range(100):
        mobs.spawn(CHICKEN, pos(0, 10, 0))
    player.on_chat("chicken", on_chat)
```

## Sequence

a programmed (coded) algorithm



## String

a linear sequence of characters, words, or other data types

```
1 agent.give("bamboo", 64,1)
2 agent.move("forward")
3 agent.place(1, "back")
```

## Test

the process of running a program to see if it is correct

