

MINECRAFT EDUCATION



ARTEMIS: RETURN TO THE MOON

EDUCATOR GUIDE

This Educator Guide includes:

- Theme Overview
- Ways to Use this Learning Content
- Multiplayer Support
- Lesson Activities
- Extension Activities
- Minecraft Visual Glossary
- Getting Started: Minecraft Education
- Educational Standards

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

THEME OVERVIEW

Welcome space explorers – It is time for launch!

Artemis: Return to the Moon is part of a series designed as a fun and creative introduction to the Artemis Program, the robotic and human exploration program led by the United States' National Aeronautics and Space Administration (NASA) and other partner agencies to return astronauts to the lunar surface. This one-hour experience will provide students, ages 8-18, with special in-game instruction to journey to the Moon using code to solve microtasks to navigate from the Earth to the Moon and back! Players will work on Artemis missions aboard the Orion, perform special missions using MakeCode blocks or Python Notebooks, and experience life as an astronaut on the Orion. This game has the ability to be played as a single player experience or as a multiplayer experience with up to 4 players (see the “Multiplayer Support” on page 7 for more information).

This Educator Guide is designed to provide you with information to help you support game play for all students. This guide will provide you background information and helpful information to help you feel comfortable and confident to utilize this learning content in your educational environment. Within this guide, you will find the necessary materials to lead an intentional experience to engage in dialog around the Artemis program, facilitate a successful in-game experience, and support student learning throughout game play.

If you are ready, accept your mission to the Moon and get started!

HOW TO GET STARTED – MINECRAFT EDUCATION

Minecraft Education offers a unique learning platform where students can engage in creative builds and various game elements.

If you are licensed to use Minecraft Education through your O365 EDU account, download directly at aka.ms/download or reach out to your IT department for assistance. More information on deployment and license assignment can be found at aka.ms/meedeployguide.

Here's how to get setup:

1. First, [check here](#) to see if your school account is eligible.
2. If you do not have a valid O365 EDU account, you can still [download](#) and try a free demo on Windows, Mac or iPad.
3. [Download](#) Minecraft Education for Windows, Mac or iPad.

Once you are set-up with Minecraft Education, you should:

- Learn about Artemis: Return to the Moon with this video: aka.ms/ArtemisMissionsTrailer
- Use this Educator Guide!
- Play through Artemis: Return to the Moon to get a better understanding of the lesson and how to navigate through the Minecraft world.

WAYS TO USE THIS LEARNING CONTENT

How can this lesson be used?

Integrated into computer science classrooms	Artemis: Return to the Moon is connected to numerous computer science standards. You could easily look at your academic standards and use this learning content when you are algorithms and programming, how to collaborate with peers to complete tasks, how to use critical thinking to solve problems, etc.
STEAM Labs or Digital Makerspaces	Who wouldn't love to use Minecraft in a fully integrated learning environment where everything from the furniture and technology to curriculum and assessment work together to support hands-on, minds-on learning?! The Artemis learning content could be used for student exploration through this applied technology.
Afterschool learning opportunities	After a full instructional day, students need something fun and engaging to provide motivation for learning. What could possibly be more fun and engaging than Minecraft Education?! Not only will students love playing Minecraft, but they will also learn educational concepts and develop their skills in creativity, communication, collaboration, and critical thinking.
Summer camps	Are you planning or hosting a summer camp? Who doesn't love a space-themed camp?! This learning content could be included in a space exploration week of fun and learning! Students will also have the opportunity to learn about great careers at NASA!
Homeschool groups	You don't have to be exclusively at a school or part of a school district to utilize Minecraft Education. Homeschools, families, and other organizations can purchase and manage Minecraft Education through our commercial offer. You can find more information here .

ARTEMIS II: RETURN TO THE MOON — LESSON ACTIVITIES

Overview of the Activities

Welcome to Artemis: Return to the Moon! When the game starts, players will be prompted to make some game play decisions. These choices are important as they will affect how the game is played.

Players will be prompted to select the type of device they are using to play: keyboard or touch controls.

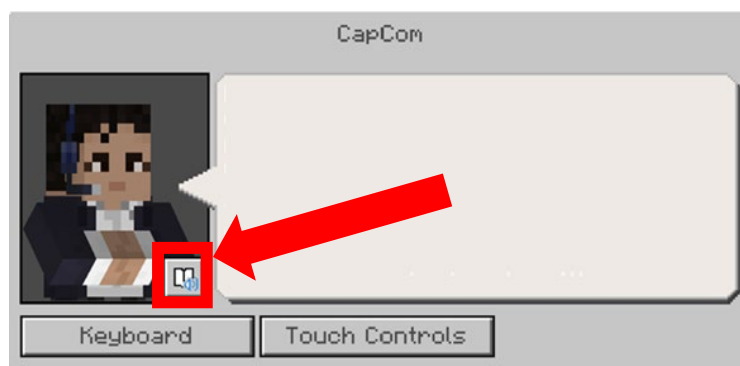


Next, players will need to select their coding language: blocks or Python. Blocks are best for beginners or players who may have never coded before. You should only choose Python if you have coded in it before.



Helpful Hint – How to use Immersive Reader

Immersive Reader is an integrated feature of Minecraft Education. It supports learners in reading or translating in-game text, including the character dialog. When a dialog box is opened, you will see an icon in the bottom right-hand corner of the NPC picture in the dialog box.



After players have selected their device configuration and coding language, they will be prompted to wait until their entire crew has logged into the world before running the code to signal they are ready to begin the game.

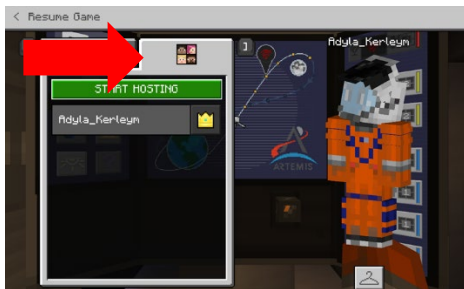
If the player is going to experience the game independently, they may continue on and run the code to begin the game.

If students are going to have a multiplayer experience (as teams of 2, 3, or 4 players), they will need to wait until their teammates have joined the game to run the start code.

Helpful Hint – How to use Join Codes to Host Multiplayer

If you are hosting the game, follow these steps:

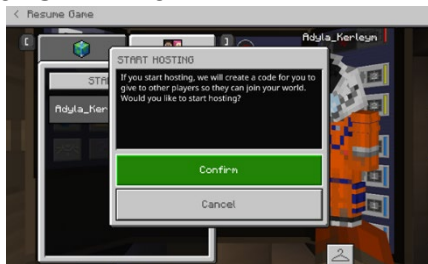
Step 1: Pause the game. Click on the Friends tab.



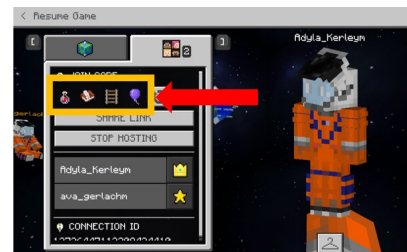
Step 2: Click “Start Hosting”.



Step 3: Click “Confirm” to start hosting.



Step 4: Provide the join code to other players who will be joining your team.

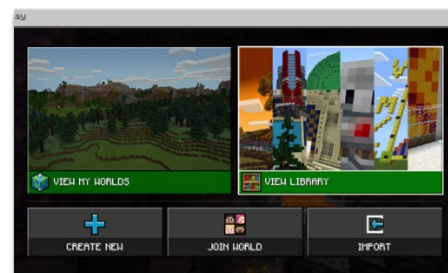


If you are joining the world, following these steps:

Step 1: Select “Play”.

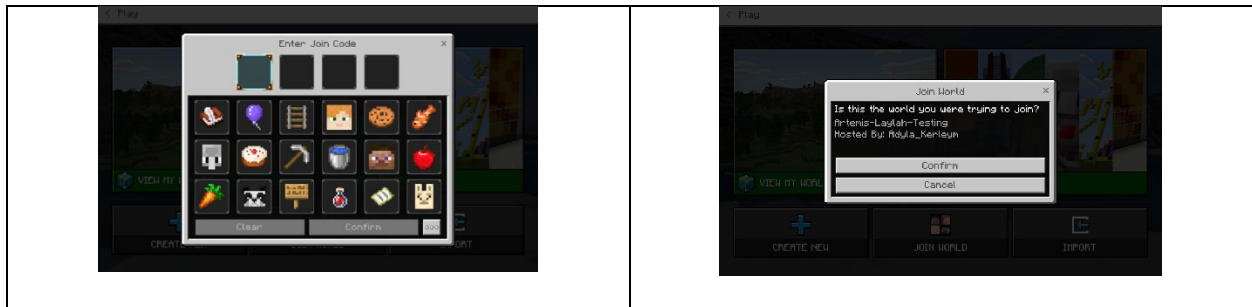


Step 2: Select “Join World”.



Step 3: Enter the join code.

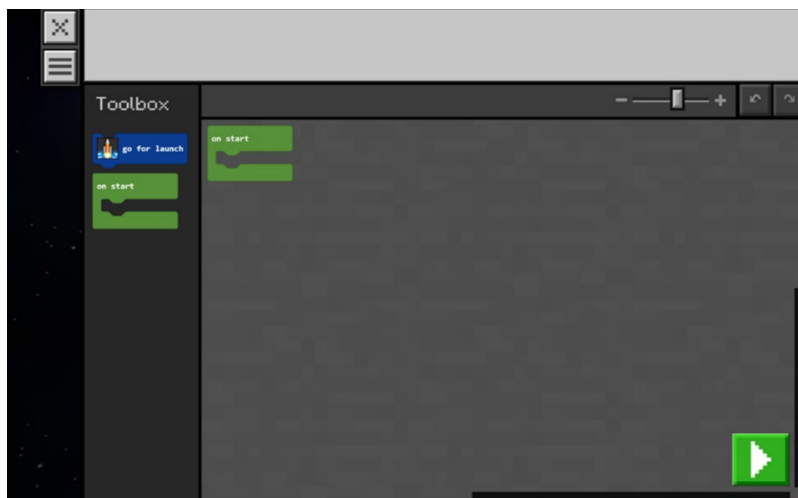
Step 4: Confirm that it's the correct world you are trying to join.



Once players join a world, they should be able to see each other on their screens and their names under the join code.




Players will now press the “C” button to open Code Builder. After players press the “C” button, Code Builder will automatically open. Code Builder is in the in-game coding editor.



Coding Solution

Blocks	Python
--------	--------

on start  go for launch	go_for_launch()
---	-----------------

Once completed successfully, players will see the cinematic scene of the Artemis II launch.



Wow! What a launch...

On the first game play, all students should select “Yes” to be shown how to use the Orion Crew Module console.



Players will need to select the flashing yellow console button.

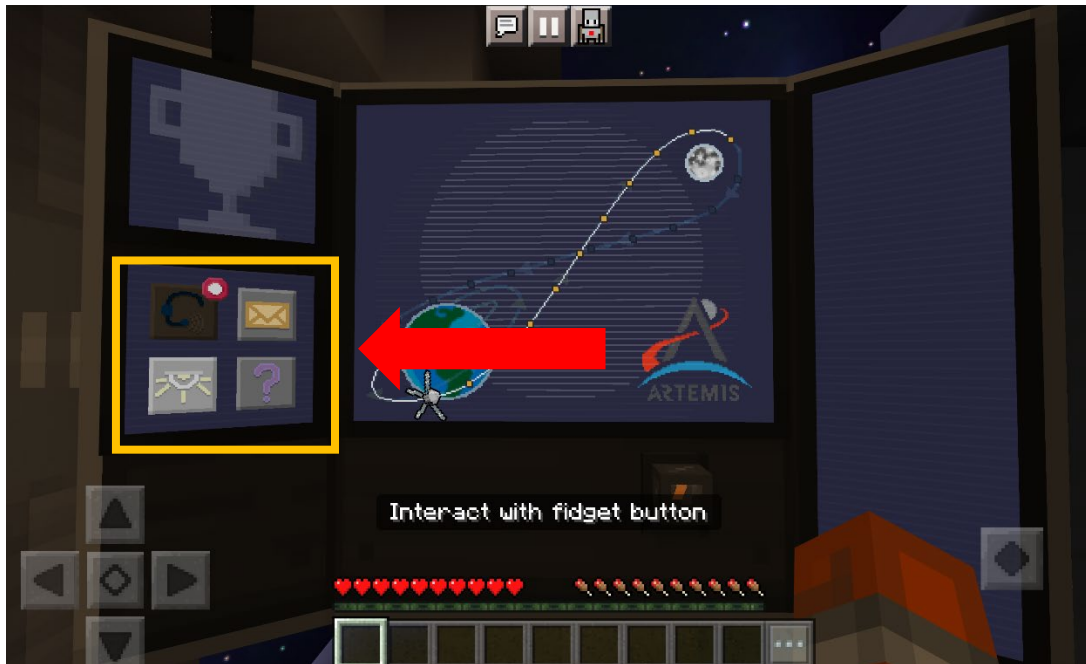
Keyboard	Touch



Interacting with this button will open the console. This console is your main screen to track your journey. The further along the Orion is, the closer you'll be to end of the mission!



Next, players will interact with the fidget budget to open the fidget screen.



The fidget screen is used for non-mission critical tasks, such as checking your email, turning off the overhead light, or chatting with CapCom!

Next, players will need to select one of the blinking task buttons (shown on the right of the screen).



As a team, the objective of your mission is to get as many points as possible before your splashdown back on Earth! The trophy icon shows how many points your team will earn for completing the task. The example shown below is worth 1 point.



Some tasks require 2 crew members, and some special tasks will require the whole crew! The icon shows you how many crew members are required to start the task. For example, the task shown below requires one crew member.



Below the main screen is your ready switch. You'll interact with the switch when you are ready to start the selected task.



MISSIONS OVERVIEW

If you want to have students play together, this game, Artemis: Return to the Moon, supports a multiplayer experience.

Multiplayer is one of the most popular and effective ways to leverage Minecraft in the classroom. It's a great way to encourage collaboration, communication and problem solving as students work together on projects in a shared Minecraft world.

Minecraft Education features multiplayer games that connect using picture-based join codes. All players in the same Office 365 tenant can host and join worlds together in the classroom. For most users, this means the part of your email address after the @ symbol matches that of the other players. A single multiplayer game can host a maximum of 40 players, and can connect players across [all supported platforms](#). **Server-style multiplayer or a dedicated server is not possible at this time.**

Here are the requirements for the best possible multiplayer experience:

- Multiplayer is possible for players within the same Office 365 tenant.
- Players should have the most [up-to-date version](#) of Minecraft Education
- All players need to be running the same version of Minecraft Education to be able to host and join worlds together.

[This article](#) will provide you with more in-depth information about hosting, join codes, troubleshooting, etc.

You can also watch [this video](#) to see how to host and join multiplayer sessions.


Although the Multiplayer experience was designed and optimized for classroom use where all players are on the same network, Multiplayer across different internet connections is possible by making some changes to your home network configuration. Visit the [How to Set Up a Multiplayer Game from Home Guide](#) to understand how to host a game that players on other networks will be able to join.

ODD PLAYER SUPPORT

Artemis: Return to the Moon is devised for a variety of game play. There are single player activities, two-player activities, and four-player activities. Here is a breakdown of what is available and supported for the number of players.

1 Player Logged In	<ul style="list-style-type: none"> Single Player Activities: Has access to everything Two Player Activities: Can do them, but will only do the "second part" of them (i.e., AI {computer} is player-one, player is player-two) Four Player Activities: Does not have access to them
2 Players Logged In	<ul style="list-style-type: none"> Single Player Activities: Has access to everything Two Player Activities: Has access to everything Four Player Activities: Can do them, but one player is communicator the second player does step 3 (i.e., AI {computer} does step 1 and step 2)
3 Players Logged In	<ul style="list-style-type: none"> Single Player Activities: Has access to everything Two Player Activities: Has access to everything Four Player Activities: Can do them, but one player is communicator the other two players will do steps 2 and 3 (i.e., AI {computer} does step 1)
4 Players Logged In	<ul style="list-style-type: none"> Single Player Activities: Has access to everything Two Player Activities: Has access to everything Four Player Activities: Has access to everything

CODING MISSIONS

SINGLE PLAYER MISSION		
Mission Title	Mission Objective	Mission Briefing
Earth's Pollution 	Launch CubeSats over cities where you see large pollution clouds.	One way to combat pollution is to understand the source of those pollutants and how they react with our atmosphere. Los Alamos National Laboratory has built a tiny satellite called the NanoSat Atmospheric Chemistry Hyperspectral

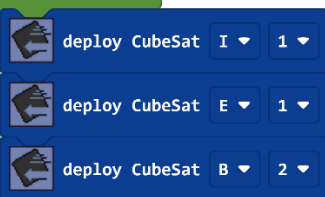
		Observation system, or NACHOS. Use NACHOS to get this important information about our atmosphere.
--	--	---



Coding Solution


Blocks

on start



Python

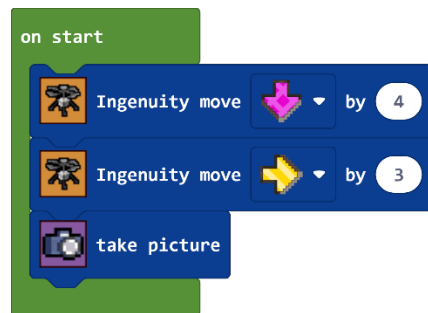
```
deploy_cubesat('i','1')
deploy_cubesat('e','1')
deploy_cubesat('b','2')
```

SINGLE PLAYER MISSION		
Mission Title	Mission Objective	Mission Briefing
Mars Recon 	Capture an image of the potential Mars landing spot.	New data has revealed a possible landing spot for our future missions to Mars. The Ingenuity Mars Helicopter, Ginny, is at our disposal. Navigate Ginny over the location and gather additional image data.



Coding Solution

Blocks



Python


```

rover_move("down", 4)
rover_move("right", 3)

take_picture()

```

SINGLE PLAYER MISSION

Mission Title	Mission Objective	Mission Briefing
Landing Site 	Mark three (3) lunar landing sites.	We found 13 places near the Moon's South Pole where we might be able to land safely. The three places should be relatively flat, have good communication with Earth, and have lots of ice nearby. Use the probe to mark the landing sites.



Coding Solution


Blocks

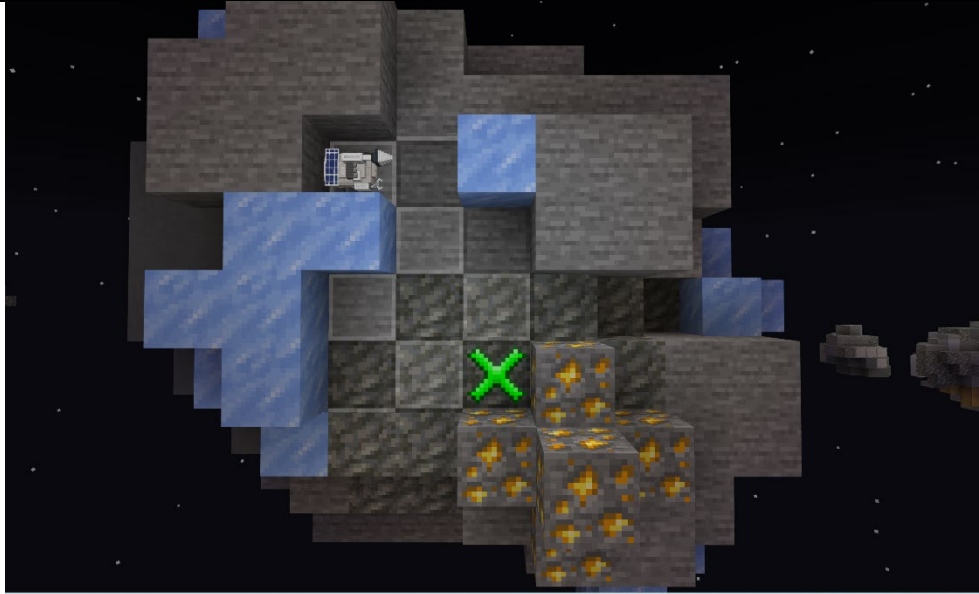


Python

```

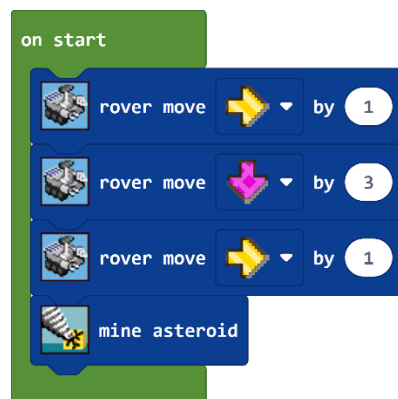
mark_location('g','1')
mark_location('f','6')
mark_location('c','5')
  
```

SINGLE PLAYER MISSION		
Mission Title	Mission Objective	Mission Briefing
Asteroid Mining 	Collect a mineral sample from the asteroid.	Astronomers have found an asteroid flying near Earth that may contain rare Earth metals, such as gold. Navigate the probe to the marked site and mine for samples to test.



Coding Solution


Blocks

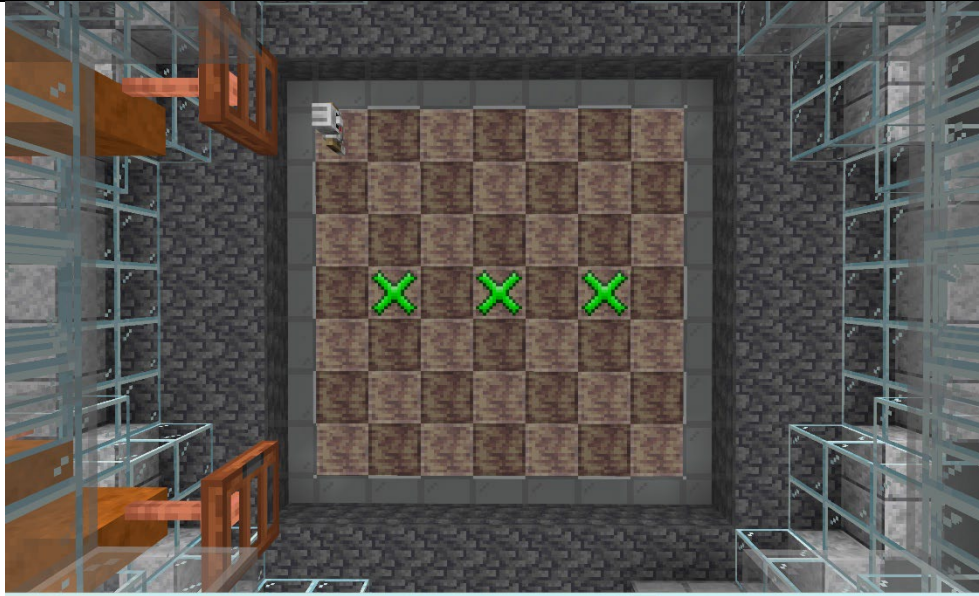


Python

```
rover_move("right", 1)
rover_move("down", 3)
rover_move("right", 1)

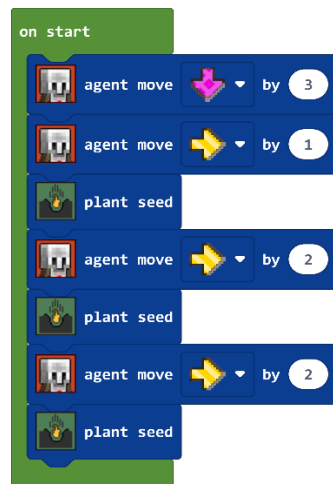
mine_block()
```

SINGLE PLAYER MISSION		
Mission Title	Mission Objective	Mission Briefing
Lunar Garden 	Plant three seeds onto the lunar regolith.	Astronauts on the Apollo mission brought back soil from the Moon, known as regolith. Scientists even grew mustard plants in regolith on Earth! Artemis missions will try growing plants on the Moon. Plant the three seeds in the lunar regolith and observe the results.



Coding Solution

Blocks



Python

```
agent_move("down", 3)
agent_move("right", 1)
plant_seed()
agent_move("right", 2)
plant_seed()
agent_move("right", 2)
plant_seed()
```

SINGLE PLAYER MISSION

Mission Title

Aurora Pictures

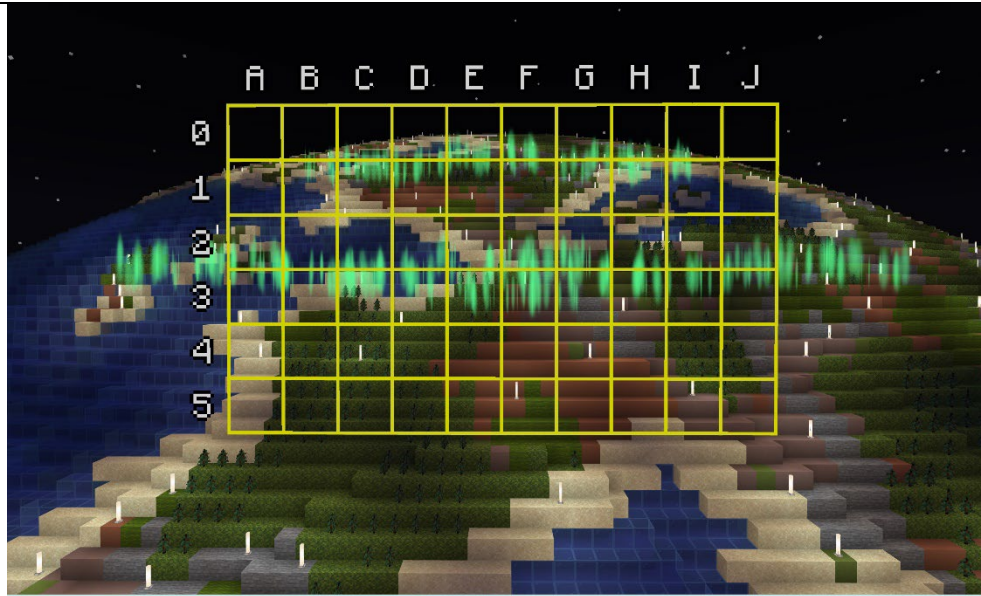


Mission Objective

Take three (3) pictures of the aurora borealis to share back to Earth.

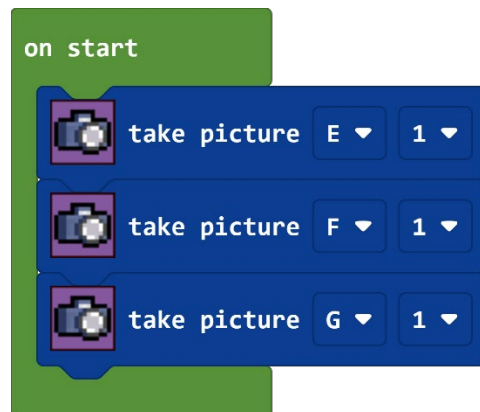
Mission Briefing

The aurora borealis, also known as the northern lights, is a beautiful show of lights in the sky, caused by particles from the Sun hitting Earth's atmosphere. Astronauts love taking pictures of the auroras from their unique perspective above Earth, so let's not waste the chance!



Possible Coding Solution
(there are multiple solutions for this coding task)


Blocks



Python

```
take_picture('e','1')
take_picture('f','1')
take_picture('g','1')
```

SINGLE PLAYER MISSION

Mission Title	Mission Objective	Mission Briefing
Space Junk 	Guide the Agent and collect three (3) pieces of space junk to complete the mission.	It is estimated that there are over 100 million pieces of debris orbiting the Earth. This space junk can be big things like broken satellites and old rockets, or small things like tiny paint flakes. Space junk moves really fast... over 25,000 kilometers per hour! If this junk crashes into other satellites or the International Space Station, it could cause big problems. NASA made a special spacecraft to clean up space junk. This vehicle is so light, eight of them can be launched in one trip!



Possible Coding Solution (there are multiple solutions for this coding task)

Blocks




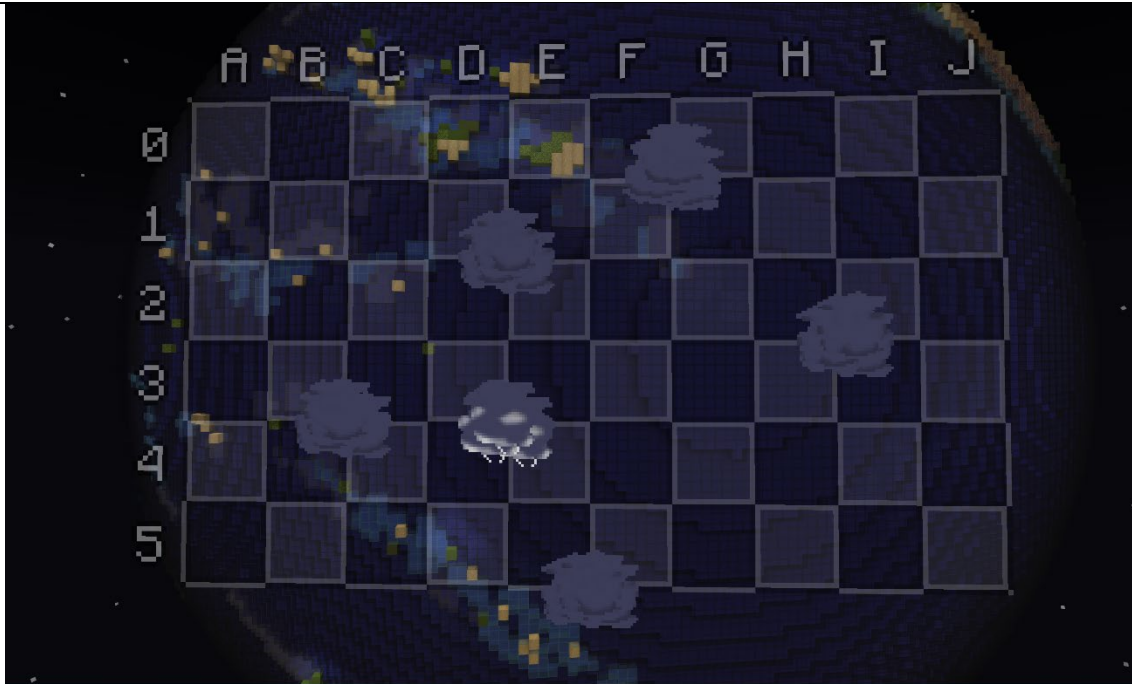
Python

```

agent_move("down", 4)
agent_move("right", 2)
collect_debris()
agent_move("right", 6)
collect_debris()
agent_move("up", 3)
collect_debris()
  
```

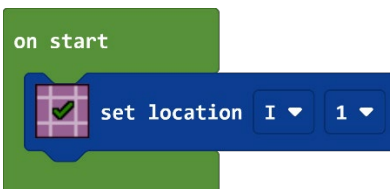
SINGLE PLAYER MISSION

Mission Title	Mission Objective	Mission Briefing
Orion Splashdown 	Choose a location for the Orion splashdown.	Several large storms are interfering with the planned splashdown location of Orion. We need to find a new location safe from thunder and lightning. Check the monitor for a location that's clear of land or storms and enter it into the landing system.



Possible Coding Solution
(there are multiple solutions for this coding task)


Blocks

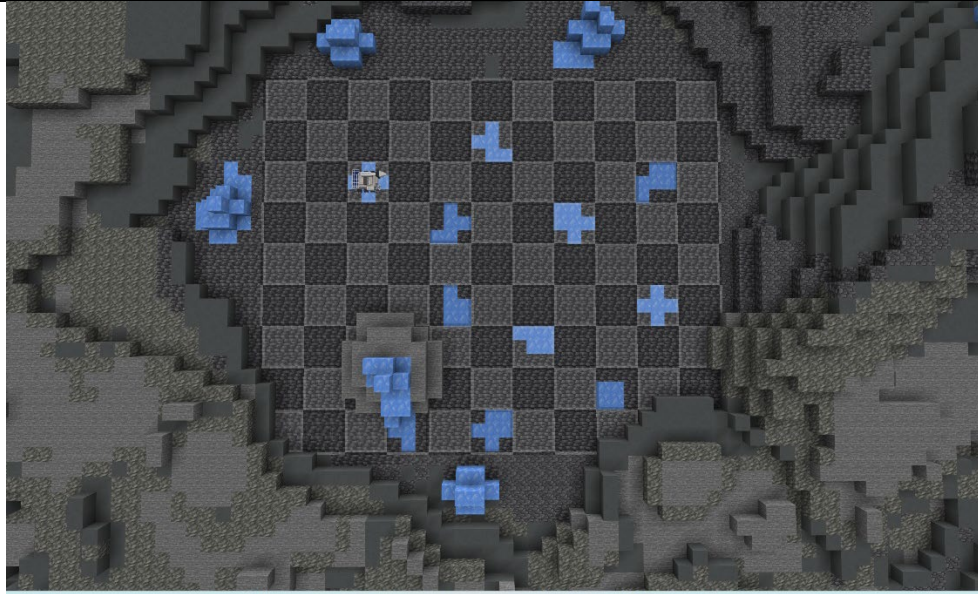


Python

```
set_location('i','1')
```

2 PLAYER MISSION

Mission Title	Mission Objective	Mission Briefing
Ice Drilling 	Locate and collect lunar ice samples.	<p>We've sent a lander to the two-mile-deep Shackleton crater on the Moon's South Pole, which is believed to hold a significant amount of ice. Use the lander to explore the area for ice. When you locate ice, use the drill to collect samples.</p> <p>Player 1 will move the rover to a location with ice. Player 2 will drill down and collect the sample.</p>



Possible Coding Solution
(there are multiple solutions for this coding task)

Blocks

Player 1



Player 2



Python


Player 1

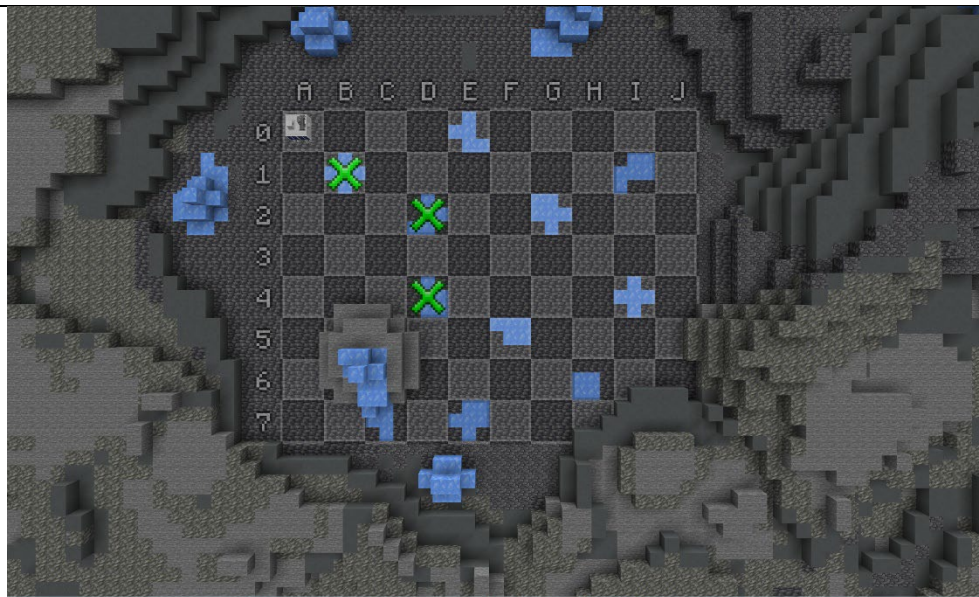
```
rover_move("right", 2)
rover_move("down", 2)
```

Player 2

```
drill_down()
collect_sample()
```

2 PLAYER MISSION

Mission Title	Mission Objective	Mission Briefing
Moon Mapping 	Mark three (3) locations on the South Pole of the Moon and navigate VIPER to each location.	Explore the Moon's South Pole using the Volatiles Investigating Polar Exploration Rover (VIPER). VIPER will scout the South Pole of the Moon for ice and other potential resources on and below the lunar surface that could eventually be harvested to sustain human exploration on the Moon, Mars, and beyond. Player 1 will mark the locations. Player 2 will move the rover and take pictures at each location.

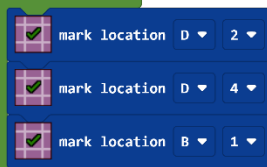


Possible Coding Solution (there are multiple solutions for this coding task)

Blocks

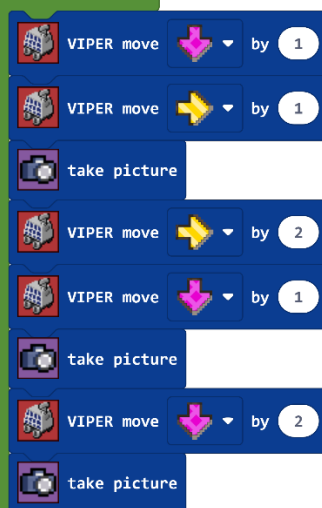
Player 1

on start



Player 2

on start




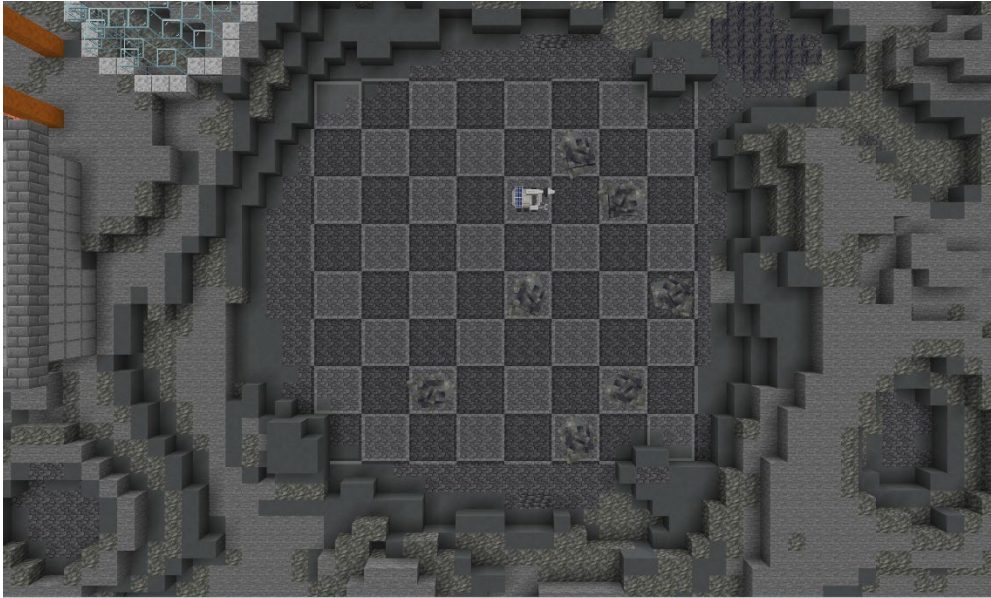
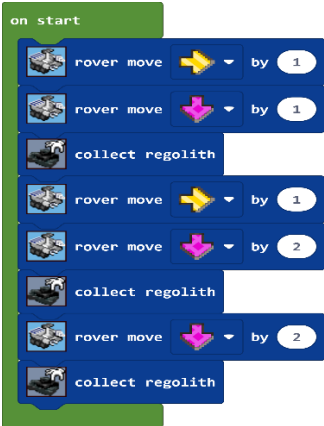
Python




Player 1


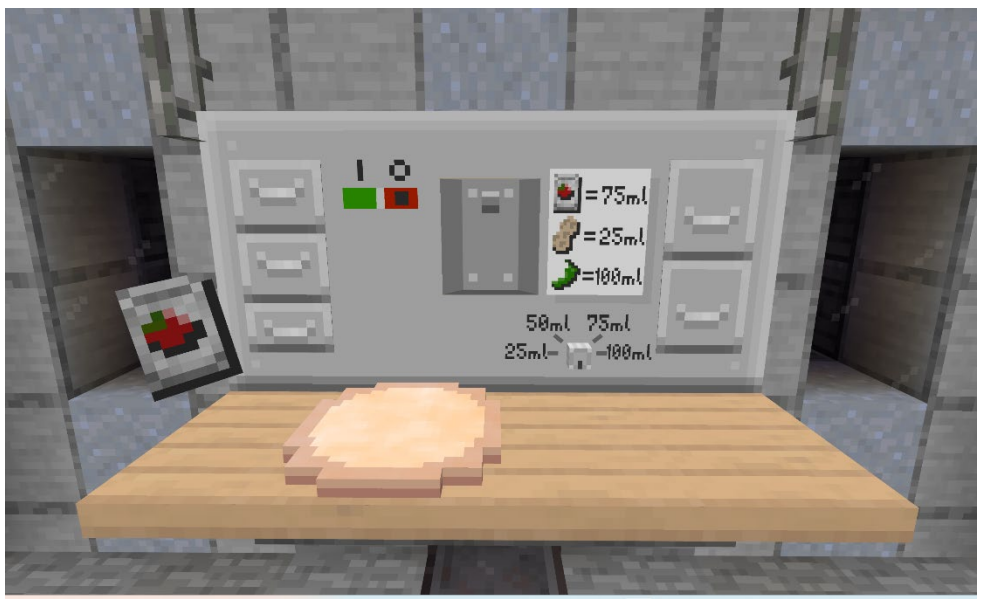
```
mark_location('d','2')
mark_location('d','4')
mark_location('b','1')
```

Player 2

```
viper_move("down", 1)
viper_move("right", 1)
take_picture()
viper_move("right", 2)
viper_move("down", 1)
take_picture()
viper_move("down", 2)
take_picture()
```


2 PLAYER MISSION		
Mission Title	Mission Objective	Mission Briefing
Building Blocks 	Collect lunar material to create building blocks.	Using the lunar materials known as regolith will reduce the cost of establishing a base and learn more about the Moon's surface and its resources. Collect lunar regolith, process it down, and form it into a block. Player 1 will move the rover and collect the regolith. Player 2 will process, cast, and test the regolith.
		
Possible Coding Solution (there are multiple solutions for this coding task)		
Blocks Player 1  Player 2	Python Player 1 <pre> rover_move("right", 1) rover_move("down", 1) collect_regolith() rover_move("right", 1) rover_move("down", 2) collect_regolith() rover_move("down", 3) collect_regolith() </pre> Player 2 <pre> process_regolith() cast_regolith() test_block_strength() collect_regolith() </pre>	

on start  process regolith  cast block  test block-strength	
---	--

2 PLAYER MISSION		
Mission Title	Mission Objective	Mission Briefing
Crew Meals 	Rehydrate the ingredients and build pizzas for the crew.	Cooking food in space is different than on Earth. Space food is freeze-dried to store and preserve it. Astronauts rehydrate the food by adding water to their meal packs. Tonight, is pizza night. With your partner, you will make the meal. Player 1 will rehydrate the ingredients. Player 2 will build the pizzas.
		
Coding Solution		
Blocks Player 1	Python Player 1 <pre> toggle_power("on") place_ingredient("tomato_sauce") rehydrate(75) toggle_power("off") </pre>	

on start

```

on start
  toggle power on
  place freeze-dried tomato
  rehydrate with 75ml water
  toggle power off

```

Player 2

on start

```

on start
  add tomato
  add cheese
  add pepperoni
  serve pizza

```

Player 2

```

add_ingredient("tomato_sauce")
add_ingredient("cheese")
add_ingredient("pepperoni")
pizza_serve()

```

4 PLAYER MISSION		
Mission Title	Mission Objective	Mission Briefing
Docking Test 	Test the docking using the NASA docking system.	The NASA Docking System (NDS) connects Orion to other spacecraft and allows the transfer of power, data, and air between them. Run a test to check if Orion can dock with the lander when it comes back from the Moon. One crew member will communicate how to set up the three (3) locking systems and instruct each crew member with their tasks.



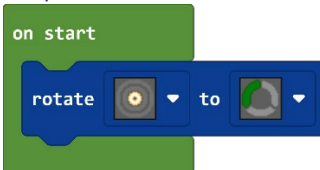
Coding Solution

NOTE: It doesn't matter which player goes first on four-player activities, but it does matter that players only run a single command each in the correct order.

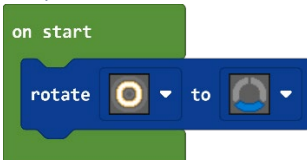
To complete the docking sequence, each ring needs to be rotated to the correct setting. Each team member should only set a single ring. Your team communicator will tell you which ring you should control and what color it should be. Players should wait for their team communicator to tell them when to run their code. Running the code too early or too late will fail the test.

Blocks

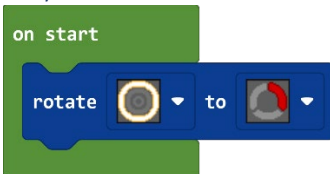
Player 1



Player 2



Player 3



Python

Player 1




```
rotate("inner_ring","green")
```

Player 2

```
rotate("middle_ring","blue")
```

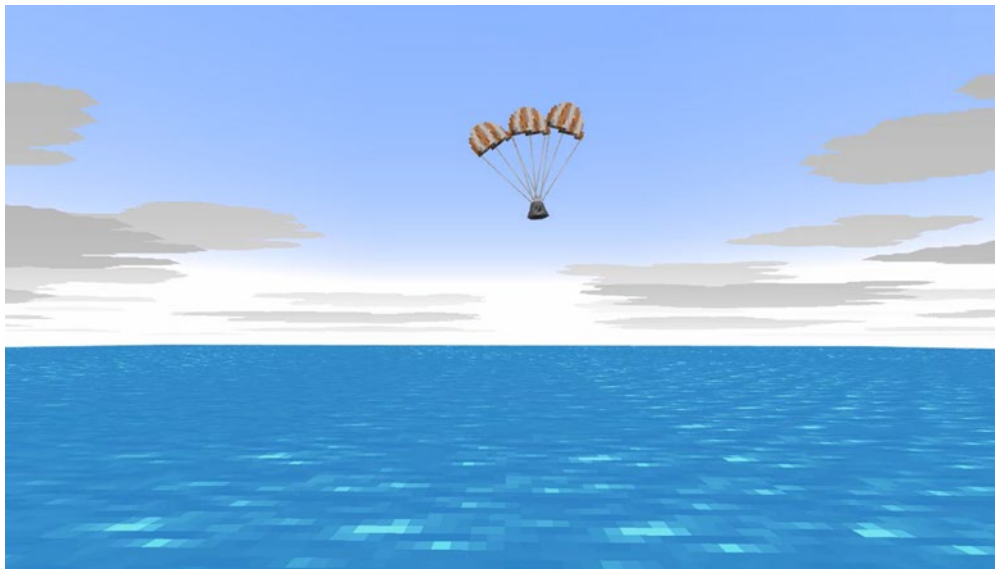
Player 3

```
rotate("outer_ring","red")
```

4 PLAYER MISSION		
Mission Title	Mission Objective	Mission Briefing
Prox Ops 	Test the Proximity Operations Demonstration Sequence	<p>One of the most important tests we planned is called the Proximity Operations Demonstration (Prox Ops). For this test, the crew will take manual control of Orion and perform a maneuver near the second stage booster. This will test the crew's ability to manually perform precise movements.</p> <p>To complete the test, the crew will work together. One of you will be the team communicator and guide the rest of the crew, step-by-step, through the Prox Ops procedures.</p>
		
<p style="text-align: center;">Coding Solution</p> <p>NOTE: It doesn't matter which player goes first on four-player activities, but it does matter that players only run a single command each in the correct order.</p> <p>To complete the docking sequence, each ring needs to be rotated to the correct setting. Each team member should only set a single ring. Your team communicator will tell you which ring you should control and what color it should be. Players should wait for their team communicator to tell them when to run their code. Running the code too early or too late will fail the test.</p>		
Blocks Player 1 	Python Player 1 <pre>move_orion("away")</pre>	

<p>Player 2</p> <pre> on start rotate counterclockwise by 360° </pre> <p>Player 3</p> <pre> on start adjust roll by 180° </pre>	<p>Player 2</p> <pre> rotate_orion("counterclockwise", 360) </pre> <p>Player 3</p> <pre> adjust_orion("roll", 180) </pre>
---	---

SPLASH DOWN! Welcome back to Earth!



Check the point system to see how well you did!

SCORE 0-8	<p>Congratulations! You've returned from your journey around the moon! You and your team have reached the rank of Astronaut! Great job! But there are many more missions to complete.</p> <p>Play again to see if you can gain more points and rank up even higher!</p>
SCORE 9-21	<p>Congratulations! You've returned from your journey around the moon! You and your team have reached the rank of Veteran! That's amazing! But you can still climb higher!</p>

	Play again to see if you can gain more points and rank up to the highest level!
SCORE 22-33	<p>Congratulations! You've returned from your journey around the moon! You and your team have reached the rank of Expert! WOW! Great teamwork.</p> <p>A perfect score would be 34 points. Play again to see if you can reach it!</p>
SCORE 34+	<p>Congratulations! You've returned from your journey around the moon! You and your team have reached the rank of Expert+! You've got a perfect score!</p> <p>Try playing with a different group of people and try out the other activities you haven't played yet.</p>

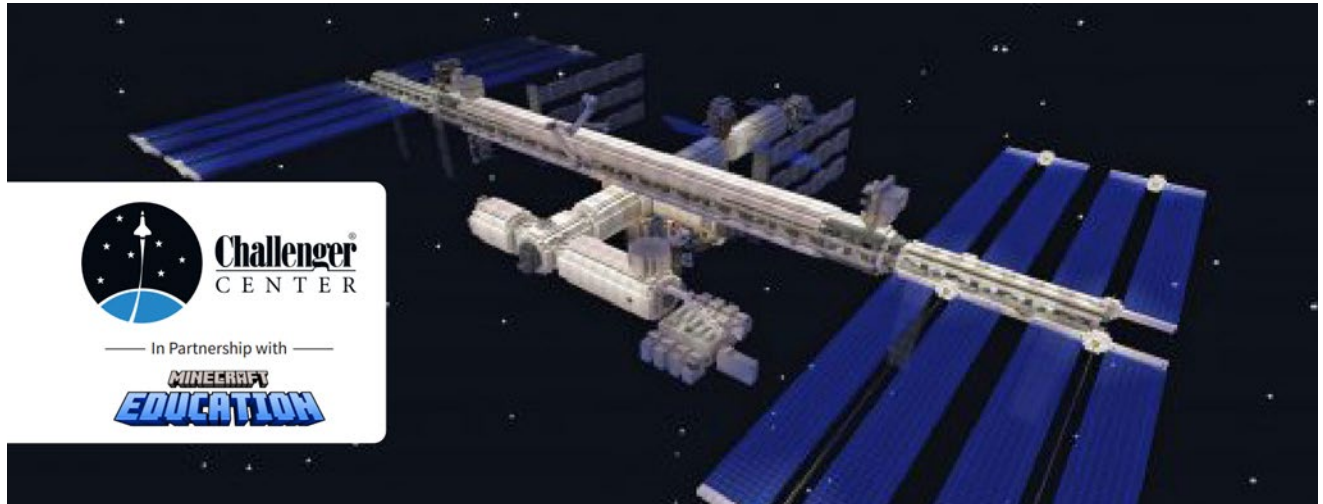
Congratulations on your lunar journey! You're part of a new generation of explorers, astronauts, and visionaries: The Artemis Generation.

Your experiences as an astronaut are just the beginning; keep your eyes towards the stars and keep dreaming big!

Players should receive their [certificate for completion](#).

EXTENSION ACTIVITIES

The learning doesn't have to stop here!



The Challenger Center has designed and created additional learning materials that fully support the Artemis missions. Use the link below to find out more information and the additional lesson plans.

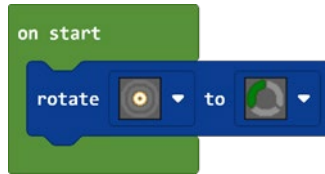
[Challenger Center Activities](#)

- Lesson 1: Apollo to Artemis History
- Lesson 2: NASA Coding and Diversity
- Lesson 3: Rockets
- Lesson 4: Moon Base Build
- Lesson 5: Teamwork
- Lesson 6: Luner Landing

MINECRAFT VISUAL GLOSSARY

BLOCKS

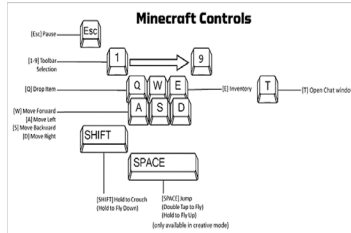
visual programming blocks used to create code



CONTROLS

(keyboard)

keyboard buttons that help you move around and complete tasks



CONTROLS

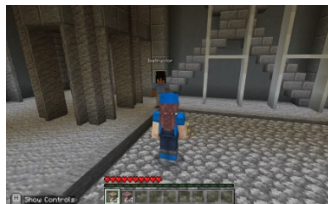
(touch)

the touch pad that helps you move around and complete tasks



PERSPECTIVE

you can change your in-game view (i.e., perspective) by pressing function + F5



DIALOG

a written conversational exchange between the player and NPC



HOTBAR

selection bar that appears on the bottom of the screen



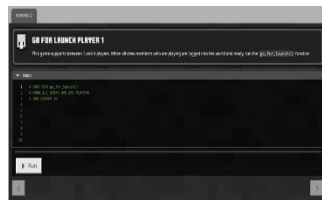
IMMERSIVE READER

a tool to help players in reading or translating in-game text



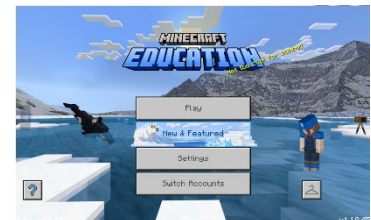
PYTHON

text-based programming language used to create code



MINECRAFT EDUCATION

a game-based learning platform



CAPCOM

non-player character who leads your through the missions



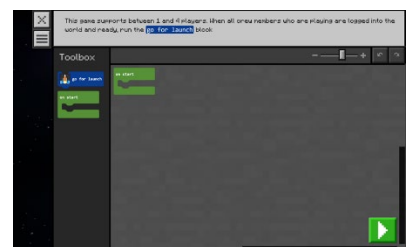
SPAWN POINT

the location where a player begins game play



CODE BUILDER

the in-game coding editor for Minecraft Education



EDUCATIONAL STANDARDS – COMPUTER SCIENCE

COMPUTER SCIENCE TEACHERS ASSOCIATION (CSTA) – UNITED STATES

CSTA Standards			
Elementary (K-2)	Elementary (3-5)	Middle (6-8)	High (9-12)
1A-AP-10 Develop programs with sequences and simple loops, to express ideas or address a problem. 1A-AP-11 Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions. 1A-IC-17 Work respectfully and responsibly with others online.	1B-AP-10 Create programs that include sequences, events, loops, and conditionals. 1B-AP-11 Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process. 1B-AP-16 Take on varying roles, with teacher guidance, when collaborating with peers during the design, implementation, and review stages of program development.	2-AP-13 Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs. 2-AP-15 Seek and incorporate feedback from team members and users to refine a solution that meets user needs. 2-AP-19 Document programs in order to make them easier to follow, test, and debug.	3A-AP-13 Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests. 3A-AP-16 Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.

EDUCATIONAL STANDARDS – COMPUTER SCIENCE

INTERNATIONAL SOCIETY FOR TECHNOLOGY IN EDUCATION (ISTE) – UNITED STATES

Computational Thinker
<ul style="list-style-type: none"> 1.5.a Students formulate problem definitions suited for technology-assisted methods such as data analysis, abstract models and algorithmic thinking in exploring and finding solutions. 1.5.c. Students break problems into component parts, extract key information, and develop descriptive models to understand complex systems or facilitate problem-solving. 1.5.d. Students understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions.
Global Collaborator
<ul style="list-style-type: none"> 1.7.c. Students contribute constructively to project teams, assuming various roles and responsibilities to work effectively toward a common goal.

EDUCATIONAL STANDARDS – COMPUTER SCIENCE

COMPUTING PROGRAMMES OF STUDY – NATIONAL CURRICULUM IN ENGLAND

Key Stage 1 <ul style="list-style-type: none"> Create and debug simple programs

<ul style="list-style-type: none"> Understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions
Key Stage 2 <ul style="list-style-type: none"> Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts Use sequence, selection, and repetition in programs; work with variables and various forms of input and output Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs
Key Stage 3 <ul style="list-style-type: none"> Understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem Use two or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions
Key Stage 4 <ul style="list-style-type: none"> Develop their capability, creativity and knowledge in computer science, digital media and information technology Develop and apply their analytic, problem-solving, design, and computational thinking skills

EDUCATIONAL STANDARDS – COMPUTER SCIENCE

AUSTRALIAN F-10 CURRICULUM – DIGITAL TECHNOLOGIES

Foundation to Year 2 <ul style="list-style-type: none"> Writing and entering a simple set of instructions jointly to sequence events and instructions (ACTDPI004)
Year 3 and 4 <ul style="list-style-type: none"> Following, modifying and describing the design of a game involving simple algorithms represented diagrammatically or in English (ACTDIP019) Planning and implementing a solution using a visual programming language, for example designing and creating a simple computer game involving decisions and repetitions, suitable for younger children, that requires user input to make selections, taking into account user responses (ACTDIP020) Experimenting with different ways of instructing to make choices and repeat instructions (ACTDIP020)
Year 5 and 6 Implement digital solutions as simple visual programs involving branching, iteration (repetition), and user input (ACTDIP02) <ul style="list-style-type: none"> experimenting with different options that involve repeat instructions planning and implementing a solution using a visual programming language, designing and creating a solution that is interactive, using a visual programming language,
Year 7 and 8

- Define and decompose real-world problems taking into account functional requirements and economic, environmental, social, technical and usability constraints (ACTDIP027)
- Implement and modify programs with user interfaces involving branching, iteration and functions in a general-purpose programming language (ACTDIP030)
- Design algorithms represented diagrammatically and in English, and trace algorithms to predict output for a given input and to identify errors (ACTDIP029)

Year 9 and 10

Implement modular programs, applying selected algorithms and data structures including using an object-oriented programming language (ACTDIP04)

- coding separate modules that perform discrete functions but collectively meet the needs of the solution
- defining classes that represent the attributes and behaviour of objects in the real world or in a game
- considering different algorithms and selecting the most appropriate based on the type of problem, for example choosing appropriate algorithms for particular problems

-