Activity Three: Navigate Your Zone

Educator Notes

Learning Objectives

Students will

- Use block programming to navigate a spherical programmable robot through a maze.

Challenge Overview

Students will learn about the basics of programming, controlling unmanned aircraft systems (UAS), and some of the many challenges NASA engineers face with the National Airspace System (NAS) as the students attempt to move a robotic ball from one side of a maze representing the NAS to the other without hitting any obstacles.

Suggested Pacing

120 to 180 minutes

National STEM Standards

Science and Engineering (NGSS)	
Disciplinary Core Ideas • MS-ETS1-1: Define the criteria and constraints of a design problem with sufficient precision to ensure a successful solution, taking into account relevant scientific principles and potential impacts on people and the natural environment that may limit possible solutions. – ETS1.A: Defining and Delimiting Engineering Problems: The more precisely a design task's criteria and constraints can be defined, the more likely it is that the designed solution will be successful. Specification of constraints includes consideration of scientific principles and other relevant knowledge that are likely to limit possible solutions. • MS-ETS1-2: Evaluate competing design solutions using a systematic process to determine how well they meet the criteria and constraints of the problem.	*Disciplinary Core Ideas (continued)* – ETS1.B: Developing Possible Solutions: There are systematic processes for evaluating solutions with respect to how well they meet the criteria and constraints of a problem. *Crosscutting Concepts* • Influence of Science, Engineering, and Technology on Society and the Natural World – All human activity draws on natural resources and has both short- and long-term consequences, positive as well as negative, for the health of people and the natural environment. – The uses of technologies and limitations on their use are driven by individual or societal needs, desires, and values; by the findings of scientific research; and by differences in such factors as climate, natural resources, and economic conditions.
Technology (ISTE)	
Standards for Students Knowledge Constructor: • 3d: Students build knowledge by actively exploring real-world issues and problems, developing ideas and theories, and pursuing answers and solutions. Innovative Designer: • 4a: Students know and use a deliberate design process for generating ideas, testing theories, creating innovative artifacts, or solving authentic problems. • 4d: Students exhibit a tolerance for ambiguity, perseverance, and the capacity to work with open-ended problems.	*Standards for Students (continued)* Computational Thinker: • 5a: Students formulate problem definitions suited for technology-assisted methods such as data analysis, abstract models, and algorithmic thinking in exploring and finding solutions. • 5b: Students collect data or identify relevant data sets, use digital tools to analyze them, and represent data in various ways to facilitate problem-solving and decision making. Global Collaborator: • 7b: Students use collaborative technologies to work with others, including peers, experts or community members, to examine issues and problems from multiple viewpoints.
Mathematics (CCSS)	
Mathematical Practices • MP.2: Reason abstractly and quantitatively.	*Content Standards* • 7.EE.B.3: Solve multistep real-life and mathematical problems posed with positive and negative rational numbers in any form, using tools strategically. Apply properties of operations to calculate with numbers in any form; convert between forms as appropriate; and assess the reasonableness of answers using mental computation and estimation strategies.

Challenge Preparation

Note**:** Make sure to plan time to find tutorials and learn more about block programming. Prior to this lesson, students should have knowledge of basic block programming. Example links: https://edu.Sphero.com/cwists/preview/21499x and https://code.org/

- Read the introduction and background information, Educator Notes, and Student Handout information to become familiar with the activity.
- Become familiar with the Disasters website from the Applied Sciences Program of NASA's Earth Science Division: https://disasters.nasa.gov/
- As needed, review concepts with students prior to the challenge (e.g., Newton's laws and calculating average or mean).
- Divide students into teams for the duration of the challenge.

- Make copies of all student worksheets for each team.
- Determine if students will use block programming or JavaScript (Oracle America, Inc.) to program.
- Prepare the NAS (challenge course) prior to the start of the lesson by doing the following:
  - Start with a large, empty space.
  - Create a NAS maze using painter's tape, polyvinyl chloride (PVC) pipe, or any available supplies.
  - Add challenges for students to complete (e.g., a circle within the NAS where vehicles pause for 10 seconds to replicate a battery recharge).
  - Add obstacles for students to avoid (e.g., a building created out of blocks within the NAS).

Materials (per team unless noted)

- ☐ Spherical programmable robot such as a Sphero® (Sphero, Inc.) robotic ball (any size)
  Note: A spherical robot best simulates the movement of a UAV due to its 360° capability.
- ☐ Tablet with a spherical robot application such as Sphero Edu® (Sphero, Inc.)
- ☐ Tape or pool noodles for NAS maze walls (per whole group)
  Note: A more permanent maze can be created with PVC pipe or similar material.
- ☐ Optional: Assorted items for obstacles and goals (per whole group)

Introduce the Challenge

Use the following challenge storyline for students, or create a storyline based on local community challenges.

> Sometime in the future… A mudslide on the base of a mountain has blocked off the only access point between a small town and the hospital 50 miles away. Due to the road blockage, the medical supplies that are shipped every Tuesday to the small-town doctor cannot be delivered. However, the hospital has just received a medical transport UAV and has offered to fly critical supplies to the doctor until first responders can clear the mudslide. Your challenge is to program and navigate the UAV through the National Airspace System (NAS) while avoiding obstacles to deliver the much-needed medical supplies. You must plan your route through the NAS prior to takeoff to ensure a successful delivery.

- Provide context for the challenge using the introduction and background information provided in the guide. Discuss the different types of UAS and the work NASA is doing to make UAS flight safer within the NAS.
- Show one or more of these introductory videos as time permits:
  - Making Skies Safe for Unmanned Aircraft. https://youtu.be/kDS-MoGVF1M
  - Videos showing UAVs in disaster areas:
    - High Tide Flooding. https://youtu.be/G-ZodfZ-mdU
    - Mississippi Flooding 2011. https://youtu.be/5ju1boh5bq8
    - NASA Surveys Hurricane Damage to Puerto Rico's Forests. https://youtu.be/HJAbGZsIjJo
- Assign no more than four students per team; two students per team is ideal if enough materials and resources are available. Inform students they will be working in the same teams throughout the challenge as they design a solution to the problem.
- Explain the details of the challenge, including the criteria and constraints.

Share With Students

Brain Booster

For over 40 years, unmanned aerial vehicles (UAVs) have been a part of NASA's fleet. These UAVs range from full-scale solar-powered versions to those using electric motors or propellers. Uses have included remote sensing for Earth sciences studies, hyperspectral imaging for agriculture monitoring, tracking of severe storms, and serving as telecommunications relay platforms.

Learn more:
https://www.nasa.gov/centers/armstrong/images/UAV/index.html

On Location

The Air Traffic Operations Laboratory at NASA's Langley Research Center provides air traffic management concept and procedure simulation capability. The system allows researchers to simulate a variety of airspace and air traffic situations to evaluate new concepts in high density traffic scenarios.

Learn more:
https://researchdirectorate.larc.nasa.gov/air-traffic-operations-lab-atol/

Criteria	Constraints
Teams must use the provided robotic ball and tablet to complete the challenge.	Teams must not touch any NAS maze wall with their robotic ball.
Teams must use block programming or JavaScript to write their program solution.	Teams must not touch their robotic ball after it has left the takeoff point.
Teams must use their own written programs from start to finish to solve the challenge.	Teams must not hit any obstacles in the NAS maze with their robotic ball.
Teams must complete all course challenges.	

- Explain any additional challenges and obstacles that have been created to make the course more difficult (optional).

Additional course challenges	Additional course obstacles

Facilitate the Challenge

Ask

- Answer any questions teams have about the challenge or their responsibilities.

Imagine

- Allow students to view the course and brainstorm ways that NASA engineers solve similar problems in the NAS.
- Ask students to brainstorm ways that their robotic ball functions like a UAV.
- Ask students to brainstorm ways a UAS might solve a problem in their daily lives.

Plan

- Have teams create a plan for how they will complete the challenge.
  - What do teams need their codes to complete?
  - In what order do their codes need to be constructed?
  - What are some potential pitfalls teams may run into during coding?
- Demonstrate for students how to find the distance traveled per rotation. This measurement will be important for planning how to navigate the course.

Create

- Teams will begin writing their codes to navigate their robotic ball through the NAS maze while meeting all criteria and constraints.

Test

- Teams may beta test their program on the NAS maze throughout the allotted time as often as time allows.
- After all allotted time has passed, teams should test their final programs on the NAS maze during their final test. Their results should be recorded in the Final Test Table in the Student Handout.

Improve

- Teams should improve and modify their program after each practice test. Remind students to record all program improvements and modifications in their plan. Encourage teams to race against themselves for time.

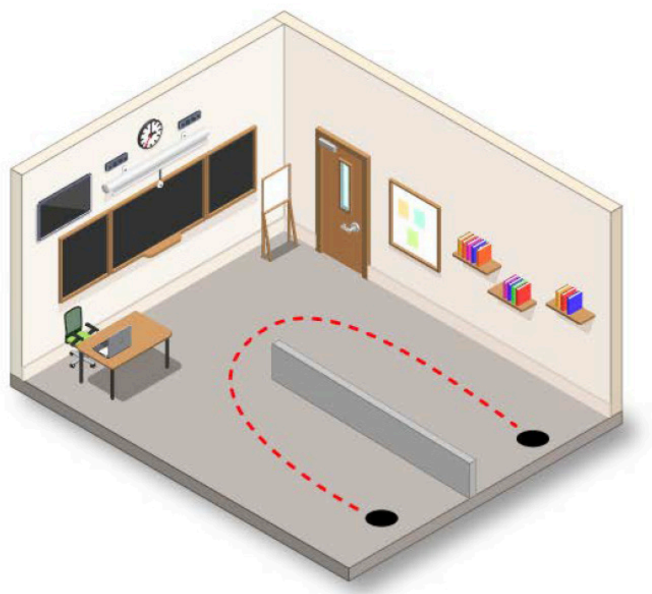
Share

- Engage students with the following discussion questions:
  - Where can you imagine UAS being helpful in your community?
  - What in your community is programmed to run autonomously (without human intervention)? (Example: traffic lights)
  - What was the biggest challenge your team faced while programming your robotic ball?
  - If you could modify your program and retest, what would you change?
- When sharing out with the whole group, teams can generate a creative way to present their solutions (e.g., news report, podcast, exhibit hall, share-out day, or allowing other teams to use their detailed coding solution to see if it works for the other teams' robots).

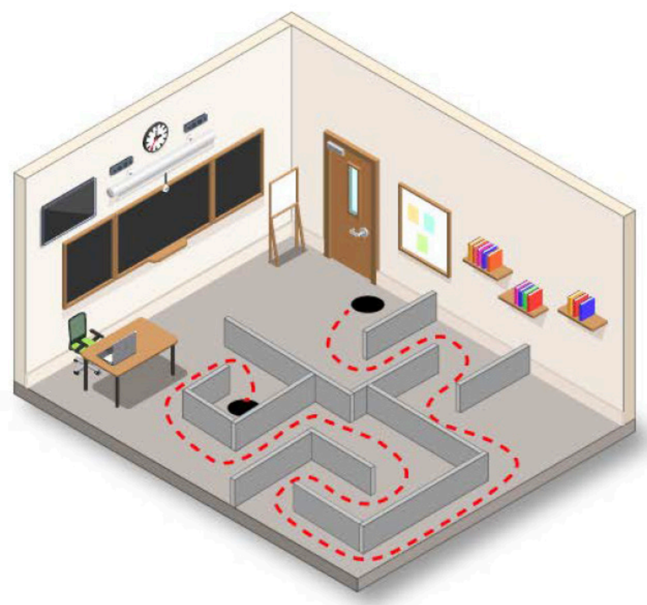
Extensions

- The NASA website on natural disasters can help students make real-world and community connections. https://disasters.nasa.gov/
- Challenge teams to create their own unique NAS maze for other teams to explore.

Example Courses



Example of a simple maze.



Example of a complex maze.

Activity Three: Navigate Your Zone

Student Handout

Your Challenge

Program your spherical robot to navigate through the National Airspace System (NAS) (maze) without hitting any obstacles.

Criteria	Constraints
Teams must use the provided robotic ball and tablet to complete the challenge.	Teams must not touch any NAS maze wall with their robotic ball.
Teams must use block programming or JavaScript to write their program solution.	Teams must not touch their robotic ball after it has left the takeoff point.
Teams must use their own written programs from start to finish to solve the challenge.	Teams must not hit any obstacles in the NAS maze with their robotic ball.
Teams must complete all course challenges.	

Additional challenges and obstacles (if any are provided by your teacher):

Additional course challenges	Additional course obstacles

Ask

Your challenge is to program and navigate an unmanned aerial vehicle (UAV) through the National Airspace System (NAS) while avoiding any and all obstacles to deliver much-needed supplies. You must plan your route through the NAS prior to takeoff to ensure a successful delivery.

Imagine

- How does a robotic ball function like a real-world UAV?
- How is your challenge to navigate the NAS maze similar to a problem NASA engineers might solve in the real world?
- How might using a UAS solve a problem in your daily life?

Plan

- Make a plan for how you will program your robotic ball to navigate the NAS maze.
  - In what order will you need to program your robotic ball to complete the challenges?
  - How will you ensure your robotic ball avoids obstacles?
- Determine the distance traveled by your robotic ball during one rotation. This measurement will be important for planning how to navigate the course.
- Use a table like the one that follows here to map out your plan for navigating the NAS maze.

Fun Fact

The National Airspace System (NAS) is like the highway transportation system. Instead of roadways, vehicles, and roadway users, the NAS includes all U.S. airspace, navigation facilities, aircraft, and airports along with all of the services, rules, regulations, policies, procedures, personnel, and equipment. In the NAS, more than 45,000 flights take off and land safely each day, totaling more than 16 million flights per year.

Learn more: https://youtu.be/gK2jDwPrDTA

Career Corner

Katharine Lee is the deputy division chief of the Aviation Systems Division at NASA's Ames Research Center. She first worked at Ames as a student intern.

Learn more about her journey to NASA: https://aviationsystems.arc.nasa.gov/about/bios/lee_katharine.shtml

Step	Description	Plan
1	Straight path 3 ft long	Use 1 movement block – Roll 0° at “50” speed for 10 s

Create

- Following your plan, write your code to program your robotic ball to navigate through the NAS maze. Keep in mind all criteria, constraints, challenges, and obstacles.

Test

- Throughout the programming process, make sure to beta test your program on the NAS maze.
- After all allotted time has passed, test your program on the NAS maze for the final test. Record your results in the Final Test Table.

Final Test Table		
Course challenges completed (+1)		
Course obstacles avoided (+1)		
Number of NAS maze walls struck (–1)		
Did your robotic ball complete the course?	Yes (+1)	No (0)
Time (tiebreaker)		
Final score		

Improve

As your team improves your code after each beta test, remember to record all program modifications in your plan.

Share

Share your team’s results with the group and answer the following discussion questions.

- Where can you imagine UAS being helpful in your community?
- What in your community is programmed to run autonomously (without human intervention)?
- What was the biggest challenge your team faced while programming your robotic ball?
- If you could modify your program and retest, what would you change?

Appendix A.—Rubrics

A.1 Engineering Design Process (EDP)

EDP Step	Novice (0)	Apprentice (1)	Journeyperson (2)	Expert (3)	Level of student knowledge (Score)
Identify the problem (Ask)	Student does not identify the problem	Student incorrectly identifies the problem	Student identifies part of the problem	Student fully and correctly identifies the problem	
Brainstorm a solution (Imagine)	Student does not brainstorm	Student generates one possible solution	Student provides two solutions	Student provides three or more possible solutions	
Develop a solution (Plan)	Student does not select or present a solution or the solution is off task	Student presents a solution that is incomplete or lacking details	Student selects a solution but does not consider all criteria and constraints	Student selects a solution that considers all criteria and constraints	
Create a prototype (Create)	Student does not directly contribute to the creation of a prototype	Student creates a prototype that does not meet problem criteria and constraints	Student's prototype meets most problem criteria and constraints	Student creates a prototype that meets all problem criteria and constraints	
Test a prototype (Test)	Student does not contribute to the testing of the prototype	Student conducts tests that are irrelevant to the problem or do not accurately assess strengths and weaknesses of the prototype	Student conducts carefully performed tests that consider one to two strengths and weaknesses of the prototype	Student conducts relevant and carefully performed tests that consider three or more strengths and weaknesses of the prototype	
Redesign based on data and testing (Improve)	Student does not contribute to the redesign	Student does not improve the design or address concerns	Student addresses one concern to improve the design	Student addresses two or more test-based concerns to improve the design	
Communicate results from testing (Share)	Student does not communicate results	Student shares random results	Student shares organized results but results are incomplete	Student shares detailed, organized results with group	
				Total	