sphero®

AP COMPUTER SCIENCE PRINCIPLES

TEACHER GUIDE

CONTENTS

INTRODUCTION

Welcome to Sphero's AP Computer Science Principles (AP CSP) unit. This unit is designed as a supplementary resource for AP CSP courses to unlock deeper engagement and provide hands-on learning for students with Sphero robots. The ten activities in this unit are aligned to the College Board's AP Curriculum Framework standards to reinforce the computational thinking practices taught in AP CSP courses and help students prepare for the AP CSP assessment. This teacher guide is designed to provide you with instructional support and the tools you need to successfully implement this unit into your AP CSP course.

Objectives

By the end of the unit, students will be able to:

- Apply computational thinking practices to design and evaluate solutions to problems
- Solve problems and complete tasks through the development of programs and algorithms
- Leverage creativity and collaboration as problem-solving strategies

Unit Structure

Throughout the unit, students will learn foundational concepts, practices, and terminology of computer science while solving problems through algorithms and programs using Sphero robots. The unit is structured into three sections based on the AP CSP Big Ideas, as outlined by the College Board. Each activity contains an opportunity for students to explore key computer science concepts, build their skills around these concepts, and bring these concepts to life through hands-on challenges and activities with their Sphero BOLT robot.

Big Idea 1: Creative Development

Students will learn how important collaboration is in developing programs and how to use an iterative design and development process in their work.

Big Idea 2: Data

Students will explore how computers handle data and how data can be used to produce new information and solve problems.

Big idea 3: Algorithms and Programming

Students will learn how to use algorithms and abstractions to create programs that solve problems and express their own creativity.

Big Idea 4: Computer Systems and Networks

Big Idea 4 is not covered in this course. We recommend you explore the following materials:

- Khan Academy - The Internet
- Code.org - The Internet

Big Idea 5: Impact of Computing

Big Idea 5 is not covered in this course. We recommend you explore the following materials:

- Khan Academy - Computing Innovations
- Code.org - Cybersecurity and Global Impacts

The unit is structured based on the "Big Ideas" of Computer Science as outlined by the College Board. We recommend implementing each activity as you see fit to correspond to the material that you are teaching in your own AP CSP course. Additionally, this guide provides recommended prerequisite activities for each AP CSP activity. We encourage your students to complete these activities in order to gain a more holistic grasp on concepts.

Pre Unit Preparation

Students do not need to have prior coding or computer science knowledge as a prerequisite for success in Sphero's AP CSP unit. However, Sphero recommends that you and your students are familiar with the Sphero BOLT robot and the basics of Sphero Edu prior to diving into the content.

Prerequisite Activities

This guide contains suggested prerequisite activities for each AP CSP activity. We recommend that you and your students complete the prerequisite activities as needed prior to beginning the corresponding AP CSP activity.

Classroom Tips

Planning for Time

The AP CSP activities are designed to fit into a 50-minute class period. However, with planning, completing the prerequisite activities, and incorporating extra challenges, the activities can extend over a few class periods.

Role of the Teacher

Sphero employs a "you learn, they learn" philosophy. This means that, as the teacher, you can think of yourself as the lead learner. You have the opportunity to cultivate open dialogue with your students to clarify questions and expectations and demonstrate flexibility in the learning process. This will foster a healthy class culture and encourage growth mindsets among your students.

Flexible Implementation

This unit is designed to reinforce and supplement what you are already teaching in your AP CSP course, so the implementation of activities is up to you! Perhaps you need a creative approach for introducing new CS topics to your class. Or, maybe you want to broaden application of specific skills. Whatever your unique situation, Sphero AP CSP can meet you where you are.

Collaboration in the Classroom

Collaboration is a key component of learning with Sphero robots. Throughout Sphero's AP CSP unit, there are many activities that have structured moments for students to collaborate with their peers and exchange feedback. Encourage an inclusive and collaborative classroom culture by pairing two to three students per Sphero robot and device.

Technology Tips

The Sphero Edu App

Download the The Sphero Edu app here.

Create Classes and Assign Activities

Create a teacher account so you can set up and manage classes as an educator. You have a few options for managing your classes:

- **Class Codes (Recommended)**

    If you prefer for students to work on assignments without a username and password, distribute Class Codes. Simply enter a Class Name and the class code will generate automatically. Give students the class code to access their assignments and continue working on their programs. Unlike standard classes, student progress is saved to the class rather than an account. Learn more here.

- **Standard Class**

    Create students accounts manually or by uploading a CSV. These student accounts include individual usernames and passwords for each student. In this model, you can assign activities to students for completion, but cannot directly assign programs. All student work is saved to their personal account and not the class itself.

- **Google or Clever users**

    You can automatically sync your classes to Sphero Edu. View more information here.
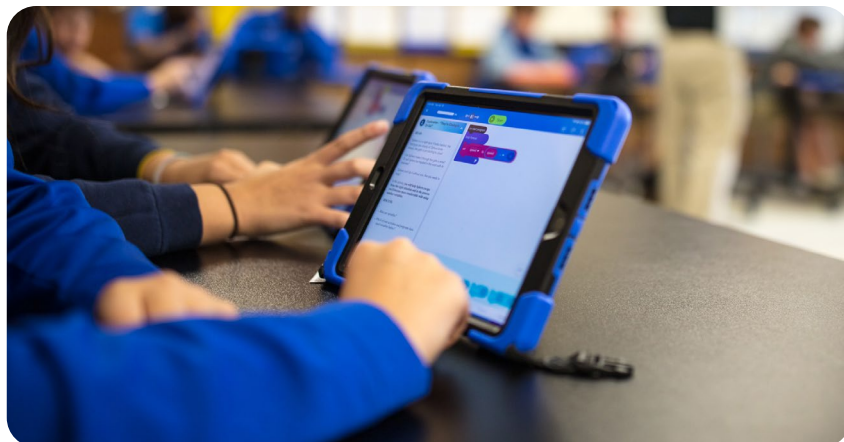
Charging Robots

Sphero BOLT robots charge via inductive charging in the provided cradle. To charge, place your Sphero robot on the charging cradle heavy side down. You will see a blue light blinking on the cradle to indicate it is charging. BOLT will need up to 6 hours for a full charge, but time will vary depending on the battery's current level and you will know it's charged when the blue light stops blinking.
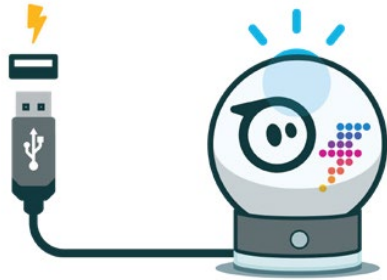
Troubleshooting

- Make sure the robot firmware is updated. If a firmware update is required, it will begin automatically after it is connected to a device.
- Charge robots and devics the night before using them in class.
- Make sure the Sphero Edu app is up to date.
- Restart your Sphero robot by holding the button down on the charger and removing the robot from the charging cradle, then place it back on the charging cradle.

Support

Sphero is empowering the future creators of tomorrow and setting them up for success. We couldn't be more excited about the future of education and the part we're playing. For more information about Sphero and to get involved in our community you can find links to additional resources below.

- **Sphero Blog -** Visit our education blog for updates, tips, and suggestions.
- **Support -** Visit our support page for FAQ and troubleshooting tips and tricks.
- **Community Forum -** Check out our community forum to connect with other educators and share resources.
- **Contact Us -** For any additional support or help contact us here.

BIG IDEA 1
Creative Development

In this section, students will explore AP Computer Science Principles Big Idea 1: Creative Development. Students will learn how important collaboration is in developing programs and how to use an iterative design process in their work.

Activity 1: Pseudocode and Flowcharts
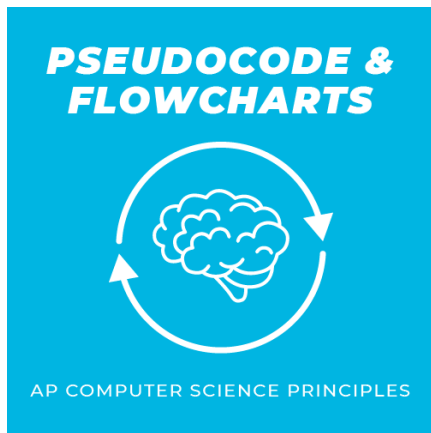
Overview

This activity introduces students to the basic logic of fundamental programming concepts using pseudocode and flowcharts. While there are many different types of programming languages, understanding how to use pseudocode and flowcharts is fundamental to developing students' computational thinking skills.

Note: Students will have access to a [pseudocode reference sheet](#) throughout the AP CS exam.

Activity Time: 50 min

Tools Needed

- Sphero BOLT
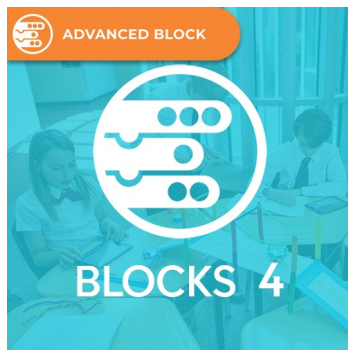- BOLT charging base connected to charging port
- Sphero Edu App

Learning Objectives and Standards Alignment

AP CSP:

- **CRD-2.I**   For errors in an algorithm or program:

  a. Identify the error

  b. Correct the error

- **CRD-2.J**   Identify inputs and corresponding expected outputs or behaviors that can be used to check the correctness of an algorithm or program.

- **CRD-2.G**   Describe the purpose of a code segment or program by writing documentation.

Prerequisite Activities

- Blocks 4: Variables

Activity 2: Program Design & Development
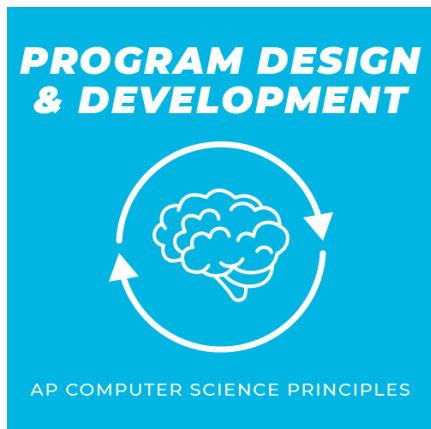
Overview

Computer programmers create and innovate using an iterative design process. This process is user-focused, incorporates implementation/feedback cycles, and leaves ample room for experimentation and risk-taking. In this activity, students will learn more about the development process and apply it to programming their Sphero robot.

Activity Time: 50 min

Tools Needed

- Sphero BOLT
- BOLT charging base connected to charging port
- Sphero Edu App

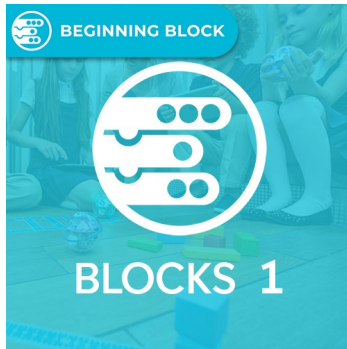## Learning Objectives and Standards Alignment

AP CSP:

- **CRD-2.E**   Develop a program using a development process.

- **CRD-2.F**   Design a program using a development process.

- **CRD-1.A**   Explain how computing innovations are improved through collaboration.

Prerequisite Activities

- [Blocks 1: Loops](#)

- [Blocks 2: If/Then, Else](#)
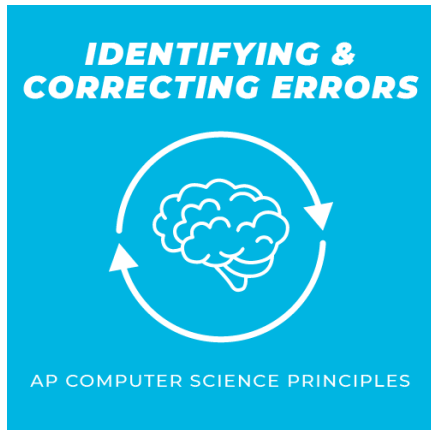

- [Blocks 3: Lights](#)

- [Blocks 4: Variables](#)

- [BOLT Matrix](#)

Activity 3: Identifying and Correcting Errors

Overview

Finding and fixing errors is an important and ongoing process in any computer science context. This activity will provide students with some strategies to debug their programs and challenge them to find and fix errors in a program running on their Sphero robot.

Activity Time: 50 min

Tools Needed

- Sphero BOLT
- Sphero Edu App

## Learning Objectives and Standards Alignment

AP CSP:

- **CRD-2.I**   For errors in an algorithm or program:
  - a. Identify the error
  - b. Correct the error

- **CRD-2.J**   Identify inputs and corresponding expected outputs or behaviors that can be used to check the correctness of an algorithm or program.

- **CRD-2.G**   Describe the purpose of a code segment or program by writing documentation.

Prerequisite Activities

- [Blocks 4: Variables](#)
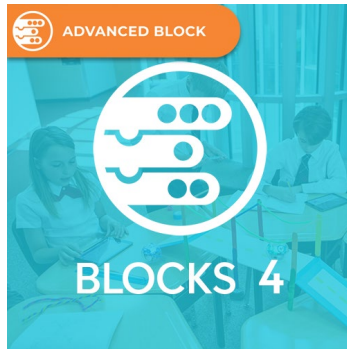
BIG IDEA 2
Data

In this section, students will explore AP Computer Science Principles Big Idea 2: Data. Data is key to computer science. Computers process, change and produce new data to solve problems and create new innovations. Students will collect, process, and analyze data with their Sphero BOLT robot in order to gain foundational computer science data skills.

Activity 1: Processing Data
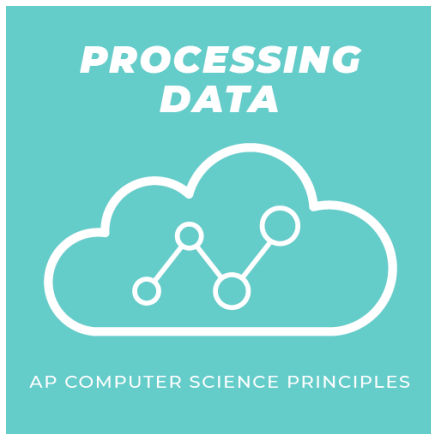
Overview

Data provides opportunities for identifying trends, making connections, and addressing problems. Programs can be used to process data, which allows users to discover information and create new knowledge. In this activity, students will learn about processing data and program their Sphero robot to provide data for analysis.

Activity Time: 50 min

Tools Needed

- Sphero BOLT
- Sphero Edu App
- Ruler or Tape Measurer

## Learning Objectives and Standards Alignment

AP CSP:

- **DAT-2.A**   Describe what information can be extracted from data.

- **DAT-2.C**   Identify the challenges associated with processing data.

- **DAT-1.D**   Extract information from data using a program.

- **DAT-2.E**   Explain how programs can be used to gain insight and knowledge from data.
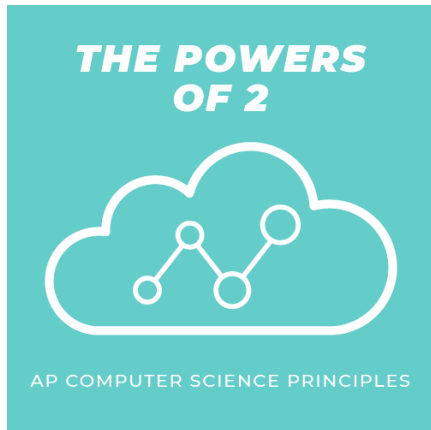
Prerequisite Activities

- Blocks 1: Loops

- Blocks 4: Variables

- BOLT Sensor Data
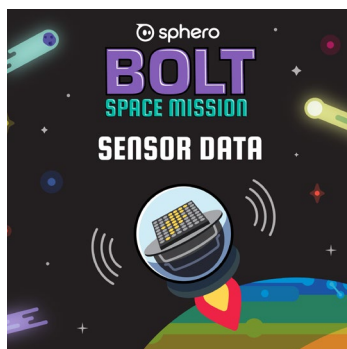
Activity 2: The Powers of Two

Overview

Programs are used to translate data into a representation more easily understood by people. The way a computer represents data internally is different from the way the data are interpreted and displayed for the user. In this activity, students will learn about how computers represent data using binary.

Activity Time: 50 min

Tools Needed

- Sphero BOLT
- Sphero Edu App

Learning Objectives and Standards Alignment

AP CSP:

- **DAT-1.A**   Explain how data can be represented using bits.

- **DAT-1.C**   For binary numbers:

  a. Calculate the binary (base 2) equivalent of a positive integer (base 10) and vice versa.

  b. Compare and order binary numbers

Prerequisite Activities

We recommend you spend time teaching your class about the binary system prior to beginning this activity.
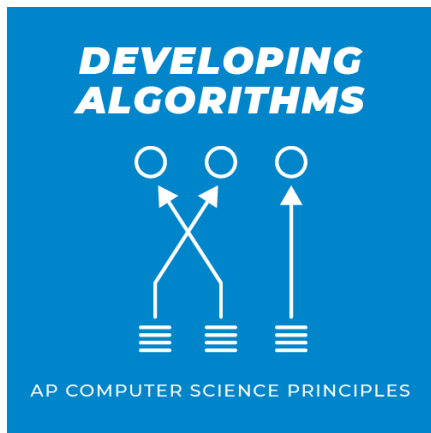
- [Jungle Code](#)

BIG IDEA 3
Algorithms and Programming

In this section, students will explore AP Computer Science Principles Big Idea 3: Algorithms and Programming. Using their Sphero BOLT robot, students will integrate algorithms to create creative programs and solve problems.

Activity 1: Developing Algorithms

Overview

Algorithms are the building blocks of programs and incorporating elements of abstraction and decomposition makes writing complex programs easier. In this activity, students will learn about the essential components of algorithms by programming their Sphero robots to dance to a song.

Activity Time: 50 min

Tools Needed

- Sphero BOLT
- Sphero Edu App
- Device that can play music

Learning Objectives and Standards Alignment

AP CSP:

- **CRD-2.E**   Develop a program using a development process.

- **CRD-2.F**   Design a program and its user interface.

- **CRD-2.G**   Describe the purpose of a code segment or program by writing documentation.

- **CRD-2.I**   For errors in an algorithm or program:
  - a. Identify the error
  - b. Correct the error

- **AAP-2.A**   Express an algorithm that uses sequencing without using a programming language.

Prerequisite Activities

- What a Character!

Activity 2: Conditionals
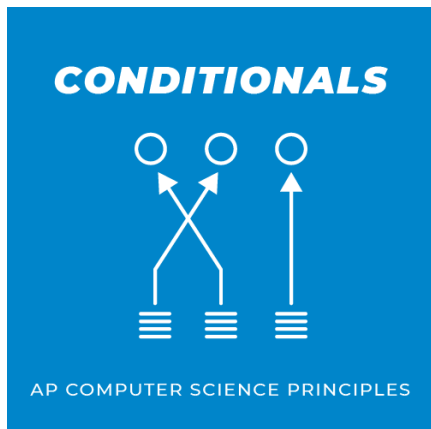
Overview

Programs use computational constructs such as loops and predefined functions to break programs up into sections. This makes programs more compact, easier to write, and easier to maintain. In this activity, students will use BOLT's light sensor to learn about conditionals and their components.

Activity Time: 50 min

Tools Needed

- Sphero BOLT
- Sphero Edu App
- Flashlight

Learning Objectives and Standards Alignment

AP CSP:

- **AAP-2.G**   Express an algorithm that uses selection without using a programming language.

- **AAP-2.H**   For selection:
  - a. Write conditional statements
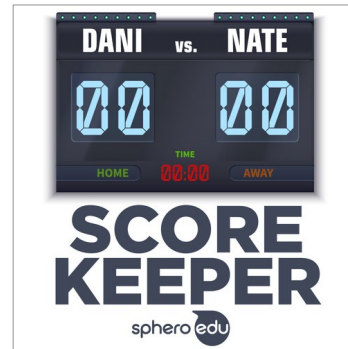 - b. Determine the result of the conditional statements

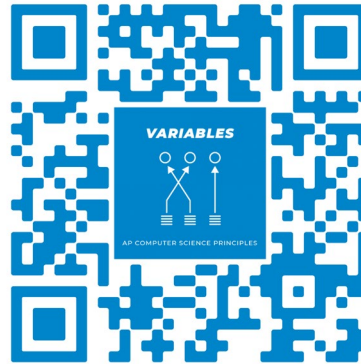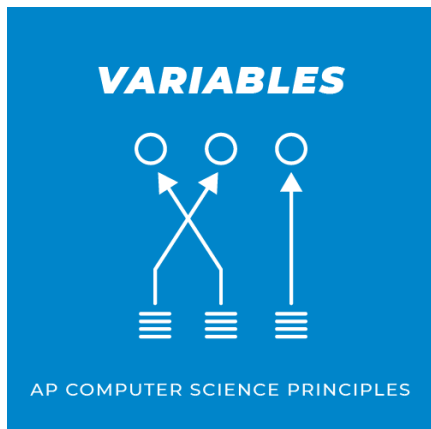## Prerequisite Activities

- [Text 2: Conditionals](#)

- [Sphero Scorekeeper](#)

- [BOLT Light Sensor](#)

Activity 3: Variables

Overview

To find specific solutions to generalizable problems, programmers represent and organize data in multiple ways. Within a program, data can be stored in variables. In this lesson, students will learn about the use of variables in programming and create an incrementing variable program with their Sphero robot.

Activity Time: 50 min

Tools Needed

- Sphero BOLT
- Sphero Edu App

Learning Objectives and Standards Alignment

AP CSP:

- **AAP-1.A**   Represent a value with a variable.

- **AAP-1.B**   Determine the value of a variable as a result of an assignment.

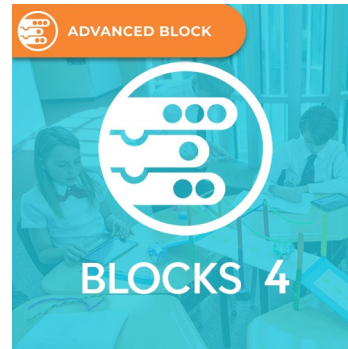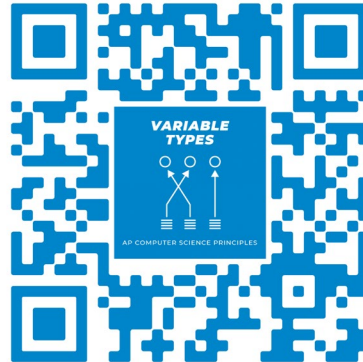- **CRD-2.B**   Explain how a program or code segment functions.
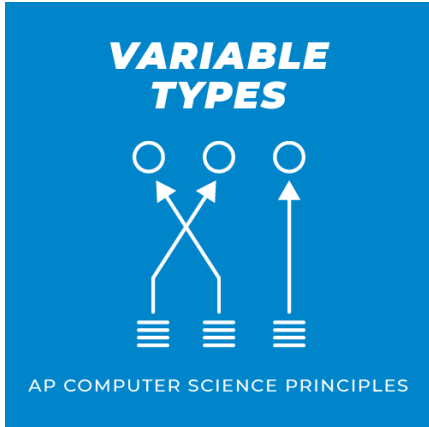
Prerequisite Activities

- [Blocks 1: Loops](#)
- [Blocks 4: Variables](#)

Activity 4: Variable Types

Overview

Programming languages like JavaScript provide types to represent data, which are referenced using variables. Within Sphero Edu, there are four types of variables: Strings, Numbers, Boolean, and Color. In this activity, students will be introduced to these data types and understand how and when they can be used in programming.

Activity Time: 50 min

Tools Needed

- Sphero BOLT
- Sphero Edu App

## Learning Objectives and Standards Alignment

AP CSP:

- **AAP-1.D**   For data abstraction:

    a. Develop data abstraction using lists to store multiple elements.

    b. Explain how the use of data abstraction manages complexity in program code.

- **CRD-2.C**   Identify input(s) to a program.

- **CRD-2.D**   Identify output(s) produced by a program.

- **CRD-2.J**   Identify inputs and corresponding expected outputs or behaviors that can be used to check the correctness of an algorithm or program.

Prerequisite Activities

- [Blocks 1: Loops](#)

- [Blocks 4: Variables](#)
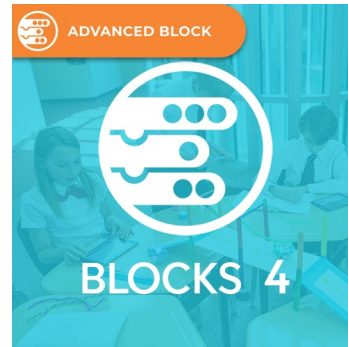
CREATE PERFORMANCE TASK

The AP Computer Science Principles Exam assesses student understanding of the computational thinking practices and learning objectives outlined in the course framework. The exam includes the Create Performance Task which requires at least 12 hours of dedicated class time for students to complete. In the Create performance task, students will design and implement a program that "might solve a problem, enable innovation, explore personal interests, or express creativity".

Sphero robots and Sphero Edu can be used to complete the Create performance task, but check with current guidelines to ensure that students are fulfilling all of the up to date requirements.

Activity 1: Innovations in Computer Science

Overview

The innovation of computer science has transformed how we communicate, process information, and conduct our daily lives. In this activity, students will explore the broad impacts of computing innovations and be challenged to think about the opportunities and limitations of computing innovations.

Activity Time: 50 min

Tools Needed

- Sphero BOLT
- Sphero Edu App

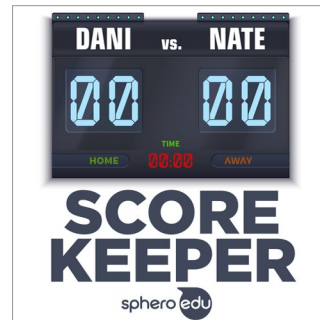## Learning Objectives and Standards Alignment

AP CSP:

- **CRD-2.A**   Describe the purpose of a computing innovation.

- **IOC-1.A**   Explain how an effect of a computing innovation can be both beneficial and harmful.

- **IOC-1.B**   Explain how a computing innovation can have an impact beyond its intended purpose.

- **IOC-1.D**   Explain how bias exists in computing innovations.

- As you work through Sphero's AP CSP activities, challenge your students to consistenting think of ways to program and invent for good.

- Refer back to the Program Design and Development activity to review Sphero's "Invention Cycle"- a framework for approaching engineering, design, and development processes.

- Students can also explore Sphero activities and programs to get additional inspiration for completing the Create performance task. For example:

  - [Lost in Space](#)
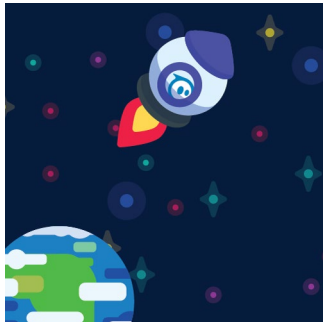
  - [Sphero Scorekeeper](#)

  - [Sphero Goes Green](#)

  - [Alarm Clock](#)