

[Title of Module]

Total Number of Lessons: [number of lessons]

Total Estimated Time for Module: [number of minutes]

SECTION 1: MODULE OVERVIEW AND CURRICULAR CONNECTIONS – REQUIRED

- **Module goal:** At the end of this module, I want students to understand how computers think and how to do some basic programming with this knowledge.
- **Connection to standards:** [List standards that will be addressed by the module. Should include Indiana standards; NGSS, CS Framework and CCSS are strongly recommended; CSTA could also be useful for the Computing RET.]
 - 6-8.CD.4: Describe what distinguishes humans from machines, focusing on ways we can communicate, as well as ways in which computers use models of intelligent behavior (e.g., robot motion, speech and language understanding, computer vision).
 - 6-8.DI.4: Create visuals such as flowcharts, diagrams, and pseudocode to represent complex problems as algorithms.
 - 6-8.PA.1: Design and iteratively develop programs that combine the following: sequencing, looping (including nested loops), conditionals (including compound conditionals), expressions, variables, functions, and parameters.
 - 6-8.CD.1: Design projects that combine hardware and software components to collect and exchange data.
- **RET materials/ideas to be leveraged:** [Describe what elements of your summer experience you will use as part of the module for students.]
 - Coding 101 lessons

SECTION 2: OVERVIEW OF MODULE FRAMEWORK – REQUIRED

- **Real-world context:** [Describe in 1-2 paragraphs]
 - **Possible Lesson Ideas:** [What are some ideas for a lesson or two focused on the real-world context and framing the project for students?]

Most people have struggled with computers not doing what they want. Have you ever tried to look something up on Google and no matter what you did, it wouldn't show you the information that you wanted. In some ways, computers are so much smarter and more capable than us humans, AI can do so much very quickly that even college graduate students are still using it to help them. However, computers can also be very stupid. I want you to learn how computers think so that you can properly utilize technology and be in command of it rather than it commanding you. We will do this by learning computational thinking, flow charts, block

coding, and then utilizing our coding skills to command hardware into doing exactly what we want.

- **Background STEM content:** [Describe in 1-2 paragraphs]
 - **Possible Lesson Ideas:** [What are some ideas for a lesson or two focused on the background STEM content?]

The idea of this lesson is how computers work on a basic level that students with no coding understanding can then begin to learn how computers think and work. I would like to see them utilize basic learning of flowcharts that they may understand, and then learn to read lines of basic code and make educated guesses as to what the code will do. I also want them to be able to see it in work as a Sphero moves based on code they create.

- **Description of Main Activity/Project:** [Describe in 1-3 paragraphs]
 - **Possible Activity/Project Ideas:** [What is the goal? What will students do? What is the timeline? At what point would students work in groups or individually?]

The goal of this lesson is to get the students to utilize computational thinking in being able to code Sphero Robots to complete various tasks. I want them to understand how computers think and process information. I would love them to know about inputs, outputs, processing, and memory. Once they are able to explain how it works, they will then learn to put code into a program and see how it works.

Students will work individually through flow charts, explaining how to do different tasks. Then when they start the coding process, I want them to learn the basics that we learned in Scratch Coding and have them show that they know how computer code makes students perform the outputs they want. I would want them to both be able to create code, but then also explain how it works. After they are able to create code in Scratch, I would then introduce Sphero Robots to the students. I would want them to be able to apply their knowledge of coding to the Sphero program and control the Sphero to move around as they plan. The Sphero program also can be very good because for excelling students, it is also compatible with Python programming and I can challenge them further.

SECTION 3: MODULE SEQUENCING AND ASSESSMENT – REQUIRED
--

- **Brief description of sequenced learning objectives:** [How do learning objectives build from lesson to lesson to meet the module goal?]
- **Computational Thinking:** 6-8.CD.4: Describe what distinguishes humans from machines, focusing on ways we can communicate, as well as ways in which computers use models of intelligent behavior (e.g., robot motion, speech and language understanding, computer vision).
 - Learn Vocabulary of Computers: Inputs, Outputs, Processing, and Storage
 - Binary Code, microprocessors, and millions of computations.
- **Flow Charts:** 6-8.DI.4: Create visuals such as flowcharts, diagrams, and pseudocode to represent complex problems as algorithms.
 - Flowcharts and [Computers do exactly what you tell them to do.](#)
 - Pseudocode and the language of computers.
- **Block/Scratch Coding:** 6-8.PA.1: Design and iteratively develop programs that combine the following: sequencing, looping (including nested loops), conditionals (including compound conditionals), expressions, variables, functions, and parameters.
 - Students will learn Scratch programming
 - Students will relate their flowcharts and pseudocode with actual code.
- **Sphero Coding:** 6-8.CD.1: Design projects that combine hardware and software components to collect and exchange data.
 - Students will then use the coding knowledge they have to code a Sphero
 - Students will predict what their code will actually make the Sphero do.
 - Students will explain possible reasons for discrepancies.
- **Brief description of formative and summative assessment approaches:** [When/what techniques will you use to assess student progress towards objectives during and at the end of the module?]
- This has 4 real divides in learning about this topic. At the very beginning, I'd want to use an assessment of their prior knowledge of these topics. By the time they are in middle school, every one of the students has used AI and understands a few things, but a lot of it is still pretty mysterious to them. Then once they finish the first section on Computational Thinking, I would want to do a summative assessment to make sure that all the students can identify inputs, outputs and processing. I do not need them to fully understand binary, but to see how it is the way computers think. Then after teaching about the flow charts, students will turn in multiple flow charts for me to see how well they understand how to create them and are able to get the outcome that they want. After this, I will want to check-up on students' understanding of coding in both Scratch and Sphero coding. I would like to utilize projects for them to turn in for this.

- **Recommendations for implementation:** [Describe any “safety tips” or advice you have for other educators who might implement this module, including logistical considerations.]
- Jumping straight to using Sphero Bolts can cause some kids to only want to drive them around and not to actually program.
- Utilize vocabulary of inputs, outputs, and processing throughout all lessons.
- Sphero Robots can be expensive. South Bend Community School Corporation has a set of 15 that you can talk with the IT department to check out for a few weeks.
- It can be best to make students self-sufficient with connecting and using Sphero Robots while also not just letting them take them out and play before they get their work done.

SECTION 4: MODULE FILES – ENCOURAGED/OPTIONAL

- **List of attached files:** [should include any lesson plans and handouts relevant to the Module, if you have them ready]
 - [File name 1 and one line description]
 - [File name 2 and one line description]
 - [File name 3 and one line description]
 - [File name 4 and one line description]
 - [File name 5 and one line description]