

# Unit 3: Computational Thinking: Programming

Content Area: **Unified Arts**  
Course(s): **STEM 3**  
Time Period: **December**  
Length: **5 Days**  
Status: **Published**

## Unit Summary

---

In this unit, students will create computer programs using loops, events, and conditionals before writing algorithms for everyday tasks. In addition, students will investigate different problem-solving techniques and discuss societal impacts of computing and the Internet. Through this, the students learn to collaborate with others meaningfully and persist in the face of difficult tasks. The complexity and depth of topics discussed are scaffolded appropriately to provide all students a rich and novel experience. At the end of the unit, students create interactive games or stories they can share.

## Student Learning Objectives

---

Students will learn to:

- create a program for a given task using sequential steps.
- count the number of times an action should be repeated and represent it as a loop.
- analyze a problem and complete it as efficiently as possible.
- employ a combination of sequential and looped commands to reach the end of a maze.
- distinguish between loops that repeat a fixed number of times and loops that repeat until a condition is met.
- use a while loop to create programs that can solve problems with unknown values.
- predict where a program will fail.
- modify an existing program to solve errors.
- identify an algorithm that is unsuccessful when the steps are out of order.
- reflect on the debugging process in an age-appropriate way.

## Essential Questions

---

- What are the fundamentals of computer programming?
- How do you create a program?
- How are algorithms used in coding?
- How can you use computer programming to complete a task?
- How can we avoid mistakes in our code? What can we do when there are mistakes?
- How does being persistent help in the debugging process?
- How does debugging relate to our everyday lives?

## Enduring Understandings

---

Students will understand that:

- computational thinking builds and enhances problem solving, allowing students to move beyond using knowledge to creating knowledge.

- computational thinking (coding/computer programming and the logic involved) is a digital tool in which all of today's internet technology is based on.

## **Application**

---

Students will be able to independently use their learning to:

- create a working algorithm.
- identify and correct bugs in code.
- decide when it is appropriate to use a loop in an algorithm.

## **Skills**

---

Students will be skilled at:

- creating working algorithms.
- identifying bugs in algorithms and adjusting the code to fix these bugs.
- choosing the most appropriate type of loop in order to solve a problem.