

ENgagement and Achievement through Computational Thinking

Prompting and Pointing Out Algorithms and Pattern Recognition Viewing Guide

Lesson 12

Topic and goals

In this Engagement and Achievement through Computational Thinking (ENACT) Algorithms/Pattern Recognition Lesson video, a teacher models how to integrate computational thinking (CT) strategies into your classroom. Framing, prompting, and highlighting are designed to empower students to take ownership of CT strategies.

The goals of the video are to support you in:

- **framing** a lesson or task that provides students with an opportunity to apply one or more CT strategies.
- **prompting** students (either verbally or using resources) as they work on a problem by applying CT strategies.
- **highlighting** examples of when and how students used CT strategies to complete their work.

As teachers become comfortable with framing, prompting, and highlighting, students will feel more empowered to take ownership of the CT strategies and integrate them into how they solve math problems.

Context

This video demonstrates **prompting/highlighting (pointing out)** in a lesson. The examples in this video will be most effective if students have experience finding the area of rectangles, including counting unit squares and using the formula $Area = length \times width$. These prior strategies support students as they compare different solution approaches and look for underlying structure. Although the video focuses on area problems, the ideas apply more broadly whenever students analyze similarities and differences across problem-solving approaches.

The four chapters in this video are listed below:

- **Chapter 1:** Prompting pattern recognition (0:55–2:29)
- **Chapter 2:** Prompting algorithms (2:30–4:20)
- **Chapter 3:** Highlighting (pointing out) pattern recognition (4:21–6:22)
- **Chapter 4:** Highlighting (pointing out) algorithms (6:23–8:32)

The content in **Chapter 3** connects to the ENACT **Lesson 4 video**. Specifically, the Lesson 4 video provides a more detailed example of how teachers can support students in examining whether specific steps of an algorithm apply to the problem they are solving.

Questions to consider when planning:

- What are some strategies my students could use to solve the problems in this lesson?
- How might a CT strategy already be part of or add to what they are already doing?
- How might I recognize when my students are using CT strategies?
- How might I identify when it would be helpful to prompt a student to use CT strategies?



Video notes: As you view the video, icons (below) will appear, indicating content related to CT strategies, student-focused practices, pedagogy, and/or mathematics. When an icon appears, you may want to pause the video to read the associated notes in exhibit 1.



Computational Thinking

When this icon appears, the focus will be on CT strategies that are being modeled through framing, prompting, and/or highlighting. The focus is on the strategy.



Student Focus

When this icon appears, the focus will be on student-focused practices that are being used: connecting to student experiences, supporting student choice by enabling multiple approaches to problems, valuing student thinking and voice, supporting student collaboration.



Pedagogy

When this icon appears, the focus will be on the teaching techniques that use interactive teaching and student learning, and/or assessing formatively.



Mathematics



When this icon appears, the focus will be on specific math concepts that are needed for solving the problem and connecting them to previous learning, and/or observing student work.




Exhibit 1. Notes for ENACT Video: Algorithms and Pattern Recognition

In ENACT Lesson 12, a coach models how to use two strategies for integrating computational thinking (CT) into classrooms. These two strategies, called *prompting* and *pointing out*, are designed to support students in taking ownership of CT practices.

The goals of the video are to support you to:

- **Prompt** students to use algorithms and pattern recognition to help them solve area problems.
- **Highlight (point out)** when you see students using pattern recognition or algorithms independently.

| Timestamp | Topic | Notes |
|---|---|---|
| Chapter 1: Prompting pattern recognition | | |
| 02:11 |  Computational Thinking | Connect pattern recognition to algorithms: This video highlights the relationship between pattern recognition and algorithms. In this chapter, students focus on identifying and explaining patterns shown in a table. The following chapter builds on this work by connecting the formula students develop for the area of a triangle to an algorithm. Highlighting these connections reinforces that algorithms often emerge from recognizing consistent patterns across related problems. |
| Chapter 2: Prompting algorithms | | |
| 02:50 |  Mathematics | Clarify limits of visual patterns in developing formulas: While the formula $A = \frac{1}{2}bh$ works for all triangles, the pattern that compares the area of a triangle to half the area of its smallest enclosing rectangle does not hold for obtuse triangles. In these cases, the base of the triangle is not the same length as the base of the enclosing rectangle. If your curriculum includes a lesson similar to the one shown in the video, a later lesson may be useful for revisiting obtuse triangles and helping students examine why the formula still applies even when the visual pattern changes. |

| Timestamp | Topic | Notes |
|---|--|---|
| Chapter 3: Highlighting (pointing out) pattern recognition | | |
| |  <p>Pedagogy</p> | <p>Support accurate interpretation of student thinking: Restating student ideas can help make their thinking visible to the class. To avoid making assumptions, it is also important to check for accuracy by asking students whether the restatement reflects what they intended. This practice helps clarify reasoning and ensures that patterns identified during the discussion are grounded in students' actual thinking.</p> |
| Chapter 4: Highlighting (pointing out) algorithms | | |
| 07:40 |  <p>Pedagogy</p> | <p>Support explanation of algorithmic thinking: Highlighting a student's use of an algorithm and inviting them to explain their thinking as they work through each step helps make the reasoning behind the procedure explicit. This approach also provides a natural opportunity to offer feedback that supports understanding and clarifies how the steps of the algorithm connect to the problem.</p> |
| 08:20 |  <p>Mathematics</p> | <p>Examine the scope of the algorithm: The algorithm developed in this scene applies to acute triangles (with all angles less than 90 degrees) and right triangles. It does not extend to obtuse triangles because obtuse triangles cannot be partitioned into right triangles in the same way. Inviting students to find or test a triangle for which the algorithm does not work can support deeper reasoning about the conditions under which an algorithm applies.</p> |

This viewing guide is part of a series of training resources related to REL Midwest's ENACT partnership.