

ENgagement and Achievement through Computational Thinking

Prompting and Pointing Out Pattern Recognition and Debugging Viewing Guide

Lesson 15

Topic and goals

In this Engagement and Achievement through Computational Thinking (ENACT) Pattern Recognition/Debugging 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.

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?

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 focus on decimal estimation problems and will be most effective if students have experience with decimal addition, subtraction, and multiplication, as well as finding exact solutions—whether mentally, by hand, or using a calculator. Although the mathematical focus is on decimal estimation, the strategies demonstrated apply more broadly to any situation in which students use estimates to evaluate the reasonableness of exact answers.

The four chapters in this video are listed below:

- **Chapter 1:** Prompting debugging (0:54–2:41)
- **Chapter 2:** Prompting pattern recognition (2:42–4:31)
- **Chapter 3:** Highlighting (pointing out) debugging (4:32–7:54)
- **Chapter 4:** Highlighting (pointing out) pattern recognition (7:55–9:42)



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: Prompting and pointing out pattern recognition and debugging

In ENACT Lesson 15, a coach models how to use two strategies for integrating computational thinking (CT) into your classroom. These two strategies, called *prompting* and *highlighting (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 pattern recognition or debugging to help them solve decimal estimation problems.
- **Highlight (point out)** when you see students using pattern recognition or debugging independently.

Timestamp	Topic	Notes
Chapter 1: Prompting debugging		
01:18	 Student Focus	Connects to student experiences: Encouraging students to think about the specific items a person in the problem is buying can help them make sense of the situation and reason more carefully about the quantities involved. Inviting students to consider items they might purchase at a grocery store can make the context more familiar and support their engagement as they analyze and debug their solutions.

Timestamp	Topic	Notes
Chapter 2: Prompting pattern recognition		
03:24	 Pedagogy	Support sensemaking through comparison of strategies: Inviting students to first think through how they would estimate the flooring cost can help them activate their own reasoning strategies. Encouraging them to write out the steps they would use and then compare their approach to Latisha’s work supports pattern recognition by helping students identify similarities and differences in estimation strategies and better understand the reasoning behind Latisha’s estimates.
03:52	 Mathematics	Clarify mathematical notation for exact values and estimates: It is important to distinguish between situations where a calculation is exact and where a value is an estimate. The “is equal to” symbol indicates an exact result, while the “is approximately equal to” symbol (\approx) signals an estimate. Some students may be unfamiliar with the approximation symbol or may question why both symbols appear within the same solution. Addressing how exact calculations and estimates can coexist within a multi-step process can help students better interpret their work and the notation used.
Chapter 3: Highlighting (pointing out) debugging		
05:46	 Pedagogy	Supports growth through feedback: Providing students with corrective feedback creates opportunities for them to reflect on their thinking and deepen their understanding of mathematical ideas. Timely and specific feedback helps students refine their strategies and supports continued learning as they revise their work.
07:33	 Computational Thinking	Connect debugging to abstraction: This example provides an opportunity to highlight that while debugging can help students correct errors and arrive at a technically accurate solution, abstraction supports them in deciding what information is most important to focus on in the problem. Drawing attention to this distinction helps students see how multiple CT practices can work together during problem solving.
Chapter 4: Highlighting (pointing out) pattern recognition		
09:41	 Student Focus	Supports student choice: The final question in the video invites students to reflect on how they are using CT practices in their work. Providing opportunities for this kind of reflection allows students to recognize their own problem-solving approaches and choose how to describe and apply CT strategies moving forward.

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