# **Logic in Coding Discussion Guide (for use during or after reading)**

1. What are logic gates? What role do they play in computers? (Logic!, p. 4-5)
   1. Logic gates are the basic parts of an electric circuit. They can receive input information in the form of electric signals and produce a related output. Logic gates help make decisions for computers.
2. According to the text, how do logic gates work to process binary data? (Working in Binary, p. 6-7)
   1. According to the text, logic gates use binary to process data. Binary data comes in the form of *0s* and *1s*, but can be referred to as “negative” and “positive,” “low” and “high,” or “no” and “yes.” Once a logic gate receives data in binary, it is processed and causes an output that either switches a circuit on or off. An output of *0* turns a switch off and an output of *1* turns a switch on.
3. What are truth tables and how do they help us understand the logic used to process data? (True or False?, p. 8-9)
   1. Truth tables are used to visually show all the possible combinations of inputs and the correct expected outcome of each. Logic tables help us understand all the possible situations that might occur when processing data.

1. What must be true in order for an AND logic gate to produce an output? What must be true in order for an OR logic gate to produce an output? (True or False?, p. 8-11)
   1. In order for an AND logic gate to produce an output, both inputs must be *1*. In order for an OR logic gate to produce an output, either or both of the inputs must be *1*.
2. Describe the role statements and functions play in coding. (Statements, p. 14-15)
   1. A statement is a command or instruction for a computer to follow. A function is a group of statements that work together to accomplish a specific goal. In order to code a well-run program, programmers must consider how to write statements that can turn into functions specific to the goals of their program.
3. What are variables? Provide an example to show how variables are used in programming. (Variables, p. 16-17)
   1. A variable is a value, or piece of information, that can change. Variables are often found in IF-THEN logic statements. For example, when programming a video game, one variable might control how a character moves. IF the right arrow key is pressed, THEN the character moves 1 step to the right.
4. What is a condition? Provide an example to show how conditions are used in programming. (Conditions, p. 18-19)
   1. A condition is a statement that can be true or false. A program may tell a computer to run a piece of code if and only if a certain condition is true. For example, when programming a video game, adding a condition can allow a character to move with more speed. IF the space bar is pressed while the right arrow key is pressed, THEN the character will move 10 steps right, or ELSE the character will only move one step right.
5. What are loops? Provide an example to show how loops are used in programming. (Loops, p. 20-21)
   1. A loop is a piece of code that causes part of a program to run over and over again. Loops are quite useful in programming and can either continue forever or have a stopping point. For example, when programming a video game, a loop can be added to continuously move a character.
6. How do variables, conditions, and loops work together in computer programs? (Loops, p. 22-27)
   1. Although it can be helpful to think about variables, conditions, and loops separately when beginning coding, they can be combined to create more complex codes and achieve more complex goals. For example, when programming a video game, a loop, condition, and variable can be combined. IF the character touches the dragon, THEN it will lose 1 health, or ELSE its health will stay the same UNTIL health reaches 0. Combining loops, conditions, and variables can be complicated, but it can also produce interesting programs.
7. Why is it important to understand how logic works in computers? (Loops, p. 28-29)
   1. Students’ answers will likely vary.
   2. It is important to understand how logic works in computers because this allows us to design effective programs that will lead to our intended outcomes.