Algorithms & Programming: Control   Grade: 4

Standard: 4.AP.C.01

Create programs using a programming language that includes sequences, loops, conditionals, and variables that utilize mathematics operations to manipulate values in order to solve a problem or express an idea.

Essential Skills

Perform mathematical operations (addition, division, etc.) on variables in a program for a purpose such as tallying a score or keeping time (ex. If the ball crosses the line score=score+1).

Essential Questions

How can **loops** make **algorithms** more efficient and easier to understand?

Why would you use **conditionals** (if-then statements) in a computer program?

Explanation

Students should be able to write **computer programs** using a programming language with the understanding that the order (sequence) of the instructions is crucial. They should understand a variety of mechanisms to make their programs more complex and efficient. **Loops** eliminate the need to repeat code and **conditionals** (if-then statements) allow a portion of a program to be carried out only if certain criteria are met. By 4th grade, students should have the ability to utilize **variables** in their program to complete calculations, (such as area or perimeter by multiplication or addition), determine the winner of a game (confirm that the score is at a certain value, or compare scores when time reaches a certain value), or other such manipulations. Students' programs can be a means of creative expression such as animating a story, designing a game, composing music or art or an exploration concepts in science, math or social studies.

Think of this as similar to….

Hearing music begin can cause a dancer to begin a dance routine. The dancer performs the steps in a certain sequence, often repeating steps or groups of steps. If the music speeds up, then the dancer needs to make their steps faster to match the music.

Implementation Examples—What would this look like in the classroom?

Title	Description	Link	Content Connection & Notes
Music and Loops	**Grade 3**--Students are given a starter program in a programming language such as Scratch that contains code (with comments) for short musical loops. Students are challenged to create their own musical piece by combining the different loops. Students can modify the starter program so that user input sets the value of a variable that determines how many times a loop is played; a conditional can be used to determine which loop is played based on the value of the variable (if user inputs 1-5 loop combination A plays, if 6-10 loop combination B plays). **Grade 4**-- Students can manipulate the value of the variable to set the tempo of the music; for example, if the user inputs 5 the wait time between beats can be 1/5; if the user inputs 10, the wait time between beats would be 1/10. **Grade 5**--Students can use the timer to determine a set amount of time--an event--when all music stops, the cat moves around or some other change. They should be able to explain what the event is (for example, the timer has reached 1 minute) and what happens when that occurs (for example, the cat disappears)	Music and Loops ; the write up does not include the variables, conditionals and events but can be easily adapted	This lesson also aligns with **CS** AP.PD.03 and requires students to be somewhat comfortable programming in Scratch or other programming language used.
Conditionals with Cards	**Grade 3**-- Although students may not know the word conditionals, they are familiar with the concept from their everyday lives. In this unplugged lesson using a deck of cards, students write algorithms that depend on things like a card's suit, color, or number to add or subtract points. Initially, students can use simple if/then statements, but can make their algorithms increasingly complex by determining what to do if the condition is not met (if/else statements). The variable "Points" can be used to track the points earned during the game. Students can translate the algorithm to code in Scratch or another language. **Grade 4**-- Create a more complex conditional where certain cards can result in multiplication or division of the points; so, for example if the card is the Queen of Hearts "Points" is multiplied by 2; if the card is an Ace "Points" is divided in half. **Grade 5**-- Create a flowchart to model the algorithm created. Add events to the algorithm, such as if the card is a Jack shuffle and restart the game.	Conditionals with Cards	This lesson also aligns with **CS** AP.PD.04. It is unplugged but can also be adapted to have students create a computer program.

Title	Description	Link	Content Connection & Notes
Variables-- Math Chat	**Grade 4**--Students can create or modify a Scratch program where user input determines the value of variables in order to calculate the area and perimeter of a rectangle based on user input. This starter project provided is incomplete and has errors (mathematical), so students have to use their knowledge of mathematics and computer science to debug and complete the program. **Grade 5**-- Students should identify and modify the event triggers and event handlers within the program. If students add a third variable, they modify the program so it will calculate volume. Students can describe the program flow to a peer, in the comments section of their program, or in a flow chart.	Use the Scratch project Variables—Math Chat as a starter or sample project	This lesson also aligns with **CS** AP.V.01 and AP.PD.03 and **Math** 4.MD.A.3 and 5.MD.C.5b. It is part of the Scratch Encore curriculum, which is available free with registration. See pages 5-8 of the Variables Module write up for additional context.

Standard: AP.C.01 Grade: 4
These annotations are a collaboration between Maryland Center for Computing Education and the Maryland State Department of Education.