Standard: AP.PD.01 Grade: 2

Standard AP.PD.01
With guidance, create a grade level appropriate document to clarify the steps that will be needed to create a sequential **program** and can be used to check if the **program functionality** is correct.

Essential Skills
Describe the ordered steps needed to create a **computer program** in a document.
Explain the desired goals of a program in a document.

Essential Questions
Why does the order of steps matter in a **computer program**, a process or a story?
How can you describe a computer program, a process, or a sequence of events in a story?
How is a computer program like the events in a story, a series of occurrences or a process?

Explanation
Students will create a planning document to clarify the sequence of events that occur in the story or the steps that are needed to describe a process or create a **program**. The importance of sequence to the achievement of the expected outcome is an essential element of planning and students should be aware of the consequences to the expected outcome if the events occur in a different sequence. The planning document may be a storyboard, a graphic organizer, short video, or any appropriate artifact and should contain the end of the story or the expected result. Students at this stage may complete the planning process with help from their teachers. By second grade, the planning document should be used to order the steps of a **computer program.**

Think of this as similar to….
A recipe is a set of instructions in a certain order that helps you achieve your goal (a cake!) Recipes help you know what to do and when to do them--like put the ingredients in the bowl before you stir.

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

Title	Description	Link	Content Connection & Notes
Dancing Alone	**Grade K**--Students use Scratch Jr. to create a silly dance for Scratch Cat using motion blocks. Students are introduced to creating sequences of code in Scratch Jr. Students use print outs of Scratch Jr. code blocks to plan their programs. **Grade 1**--Using the printouts of the programming blocks, students should identify how the order of the motion blocks determines the order of the dance and explain what they wanted the cat to do. Students should predict how changing the order of the blocks will change the dance, explain why the reasons for the changes and the reasons they make the changes in the code that they do and test their predictions. **Grade 2**--Students should use the printouts to create an algorithm and explain what they intend their program to do. They then program their algorithm and compare the outcome with their plan.	Dancing Alone and printouts of coding blocks	This lesson aligns with **CS AP.A.01** and is similar to Getting Loopy.
Ruby's Algorithms	**Grade 1**--Students are given directions to complete tasks, beginning with familiar tasks. Then, using the algorithm activity map, students create algorithms for Ruby to visit her four friends. Students use directional arrows to show the path she would follow. **Grade 2**--Students create algorithms to complete familiar tasks and then create algorithms for Ruby to visit her friends using the algorithm activity map. Student's document which algorithm should be used to visit which friend and other students check if the algorithms function as intended. "	Ruby's Algorithms	This lesson aligns with **CS AP.A.01** and uses the book *Hello Ruby: Adventures in Coding.*
Crazy Character Algorithms	**Grade 2**--Students will create algorithms (steps needed) for drawing crazy characters. Then, they will test and refine these algorithms by challenging teammates to follow their instructions and compare. By testing the work of their peers, student will show that they understand how to check the functionality of a program.	Crazy Character Algorithms	

Standard: AP.PD.01 Grade: 2

These annotations are a collaboration between Maryland Center for Computing Education and the Maryland State Department of Education.