Computational Thinking in Film Making

Participant Name:	**Krystal Hewer**
District:	**Colton-Pierrepont Central**
Grade Level:	**Secondary**
Subject/Course:	**Film Making (easily adaptable for any subject)**
Cross-curricular Link:	**ANY: Can be tailored to any subject matter that has a step-by-step process, Home Economics, A Science Lab, A math problem, etc.**
Approximate Time (IN MINUTES):	**200 Minutes**

CONTENT AND SKILLS

Learning Objectives:

- Define an algorithm.
- Explain the concept of decomposition.
- Define abstraction.
- Understand the basic principles of video editing.
- Identify a process that can be taught to others.
- Decompose a complex process into a series of smaller, more manageable steps.
- Abstract the essential elements of the process to simplify its explanation.
- Develop an algorithm (step-by-step instructions) to perform the identified process.
- Create a storyboard to visually represent the steps of the process.
- Film the process according to the storyboard.
- Edit the filmed footage using video editing software to enhance clarity and engagement.

Essential Questions (optional):

- What is an algorithm?
- What is decomposition?
- What is abstraction?
- What is video editing?
- How can we break problems down into understandable and repeatable steps?
- How can we use video editing software to help teach algorithmic processes to others?

Students' I can statements . . .

- I can identify a process that others might want to learn
- I can break the process into easier to understand steps
- I can abstract the steps for ease of understanding for others
- I can write an algorithm to perform these steps
- I can write a storyboard to explain the steps
- I can film my storyboard
- I can edit my film using editing software (WeVideo, iMovie, Adobe Premiere or Express, etc.)

How will you meet the needs of SWD and ELL/MLL students?

NYS COMPUTER SCIENCE AND DIGITAL FLUENCY STANDARDS
List all standards that authentically align (e.g., K-1.CT.4)

- 7-8.CT.10 Document the iterative design process of developing a computational artifact that incorporates user feedback and preferences.
- 7-8.DL.2 Communicate and collaborate with others using a variety of digital tools to create and revise a collaborative product.
- 7-8.DL.4 Select and use digital tools to create, revise, and publish digital artifacts.
- 7-8.DL.5 Transfer knowledge of technology in order to explore new technologies.
- 7-8.IC.7 Explore a range of computer science-related career paths
- 9-12.CT.10 Collaboratively design and develop a program or computational artifact for a specific audience and create documentation outlining implementation features to inform collaborators and users.
- 9.12.DL.2 Communicate and work collaboratively with others using digital tools to support individual learning and contribute to the learning of others.
- 9.12.DL.4 Independently select advanced digital tools and resources to create, revise, and publish complex digital artifacts or collection of artifacts.
- 9.12.DL.5 Transfer knowledge of technology in order to use new and emerging technologies on multiple platforms.
- 9-12.IC.7 Investigate the use of computer science in multiple fields.

OTHER SPECIFIC STANDARDS (e.g., Content, SEL Benchmarks)
List all standards that authentically align
https://www.p12.nysed.gov/sss/documents/SELBenchmarks2022.pdf

INSTRUCTIONAL PLAN
List the steps of the lesson, including instructions for the students.
Add and highlight Standard Indicator next to activity that aligns

LESSON DAY (42 Min Period)	LESSON STANDARDS AND INSTRUCTIONS	MATERIALS NEEDED
1	Introduction Day: Break it Down, Keep It Simple Take students through the first section of the slideshow. Students will	Slideshow Paper Pencils
2	What is an Algorithm?	Slideshow Paper Pencils
3	Now that students created an Algorithm, today is a BUGTEST day. 1) Separate students into pairs. Students will then read and act out each algorithm EXACTLY as written. Choose a good humored student as a volunteer and bug test their algorithm where you follow it as literally as possible to find errors. 2) Have students work together to make their algorithms	Slideshow Paper Pencils

	more clear, more efficient, and streamline. 3) Bring the students back for discussion about what they learned and changed, then have the pairs join other pairs to bug test. 4) Ten minutes to the end of class, have students come together and do a quick-write of why bug testing is important and what improvements were made in each algorithm. Have them turn in their new algorithm.	
4	What is Storyboarding and how can we translate a written algorithm into visuals? Using the slideshow and video, introduce storyboarding and scripting. Compare the algorithm to a script first. Model storyboarding an algorithm as a class, specifically the kite algorithm from the first lesson. Think-Pair-Share- What are the benefits to a visual algorithm over a written algorithm? IN CLASS/HOMEWORK: Draw up a Storyboard for you to create your own tutorial video!	
5-9	WORK TIME: Using WeVideo (which may or may not be taught with this lesson or previously.) Create a tutorial video of your algorithm. You may use your Chromebook to film, WeVideo (or other software) to edit. Add music and	
10	PRESENTATION DAY Write down any potential bugs you could see Try to follow the steps in the tutorial video	

SPECIFIC NEEDS: MATERIALS / RESOURCES / TECHNOLOGY

Add additional resources needed for this lesson such as instructional technology templates, images, videos, etc.

- SLIDESHOW