CodeX Remix Task

MISSION 14: Line Art	**PROJECT: Sketch Symphony**	**# HOURS: 1**
PROJECT GOALS: Students will remix the Line Art code to create a program that can distort images.	**EXTRA PROJECT MATERIALS:** ● **none**	**VOCABULARY:** ● Envelope: functions that map one set of coordinates to another.

Project Choices (choose one or more)"

- Build a program that generates pixel art images with the ability to distort the image using envelopes.
- Create a set of predefined envelopes like a wave, a bulge, or a twist that a user can choose from.
- Use a function to distort your pixel art with envelopes.
- Design a simple user interface that includes options for drawing, selecting envelopes, and controlling distortion.

SUCCESS CRITERIA:

- ❏ Define a function to generate an image or shape.
- ❏ Create a new envelope function and apply it to a simple pixel art image.
- ❏ Create a simple user interface to select and apply different envelope functions.

RUBRIC:

CSTA Standard	Basic (3)	Proficient (4)	Mastered (5)
Documentation			
2-AP-10 Use flowcharts and/or pseudocode to address complex problems as algorithms.	Incomplete flowcharts.	Flowcharts provided for each process.	Flowcharts provided for each process. Evidence of revisions and improvements made.
Algorithms and Programming			
2-AP-12 Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.	No loops or conditionals.	Loops or conditionals used correctly in most instances.	Loops or conditionals are used correctly in each process as needed.
2-AP-13 Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.	Code is not organized or readable. No comments.	Code is sometimes organized into problems and subproblems in order to make it organized and readable. Comments used inconsistently.	Code is decomposed into problems and subproblems, making it easy to follow and read. Comments are clear and easy to understand.
Computing Systems			
2-CS-02 Design projects that combine hardware and software components to collect and exchange data.	No hardware used; hardware does not collect or exchange data correctly.	Hardware and software components incorporated; collects and exchanges data inconsistently.	Hardware and software components are incorporated; collects and exchanges data consistently.
Collaboration			
2-AP-15 Seek and incorporate feedback from team members and users to refine a solution that meets user needs.	Team members did not work together; strengths or suggestions of each member were not incorporated.	Team members usually worked effectively as a team; strengths and ideas of each member were incorporated somewhat unequally.	Team members worked effectively; the strengths and ideas of each member were incorporated.
2-AP-18 Distribute tasks and maintain a project timeline when collaboratively developing computational artifacts.	Unequal contributions from each team member; project not completed by deadline.	Somewhat equal contributions from each team member. Project completed on time, but may have needed revisions past deadline.	Team members contributed equally; project completed on time.
Debugging			
2-CS-03 Systematically identify and fix problems with computing devices and their components.	Code bugs not identified; little or no documentation of fixes.	Code bugs mostly identified and fixed; adequate documentation of fixes.	Code bugs identified and fixed; extensive documentation of fixes.