OVERVIEW:

Lesson Plan 3

CHALLENGE MASTER

Grades:	**3-5**
Group Size:	**Pairs**
Setup Time:	**5 minutes**
Total Time:	**180 minutes**
Activities:	**4**

LESSON PLAN OUTLINE

- Activity 1: Scavenger Hunt – 45 minutes
    - 3 tasks
- Activity 2: Multiple Subroutines – 45 minutes
    - 3 tasks
- Activity 3: Pseudocode – 45 minutes
    - 3 tasks
- Activity 4: Dance Off! – 45 minutes
    - 2 tasks

OUTCOMES

- By the end of this section, students should be able to:
    - Demonstrate and show understanding of the TagTiles®.
    - Create multiple subroutines embedded within other functions.
    - Write pseudocode before creating code for KUBO.
    - Create code for KUBO to meet specific criteria.

ASSESSMENT

Students can show mastery of the content by:

- Demonstrating and showing understanding of the TagTiles.
- Creating multiple subroutines embedded within other functions.
- Writing pseudocode before creating code for KUBO.
- Creating code for KUBO to meet specific criteria.

PREREQUISITE KNOWLEDGE

- Coding+ Lesson 2
    - If it has been a while since students have worked with KUBO, you might want to review vocabulary and lessons from Coding+ Lesson Plan 2.

TEACHER PREPARATION

- Have devices available for students to follow along with the slides on *www.kubo.education* or project the slides for the entire class.
    - KUBO.education > Classroom Activities > The Coding License
- Make copies of worksheets for each student.
- Make sure all KUBOs have been fully charged before beginning.

- Find an appropriate place to do the activities. KUBO can be used on a table or the floor, but the surface must be level and clean. If you're using KUBO on a tabletop, make sure KUBO doesn't fall off the table.
- Help students find the TagTiles and activity map they will need. You might want to consider hanging up one activity map in front of the whole class to use for discussions and demonstrations.
- It's helpful to show students how to properly handle and store KUBO and the tiles. Stress the importance of taking care of both KUBO and the tiles.
- Let students know it's OK to make mistakes as long as they figure out how to debug and fix the problem.
- If KUBO turns immediately after starting a route, it might help to remove the Play Function tile as soon as KUBO passes over it. This will ensure that KUBO moves correctly.
- When students create routes and functions, it is important for them to understand that KUBO has the same capabilities humans do. For example, KUBO can't drive through walls, fences, water, fire, and so forth.
- You might find it helpful to review with students what they have already learned before going on to teach the new material.
- KUBO's lights can be many colors. When KUBO is not doing anything, you should see blue. When KUBO is recording/memorizing, you should see purple. When KUBO is executing/performing, you should see green. If something is wrong, KUBO will turn red. Removing KUBO's head will clear the error. Removing KUBO's head has no effect on memory. You might want to demonstrate these colors to the class and give them tips on how to troubleshoot.

MANAGEMENT

- It is recommended the students be put in groups of two and share one KUBO kit.
- You might find it helpful to create roles for students or number them (Partner 1/Partner 2) so that each student gets a turn being in charge of KUBO.
- Have students detach KUBO's head from the body and put the tiles away in between activities or anytime you are giving instructions.
- You might also find it helpful to give students who are new to KUBO some time to free play and discover on their own so they will be more focused when receiving instruction.
- Circulate through the room and provide help as necessary. However, to encourage student-centered active learning, instruct students to follow the "ask three, then me" rule, in which they consult each other before they consult you.
- Many of the questions posed can be answered orally or written down. You will need to let students know how you would like them to be answered.
- Extension activities are not included in the 45-minute time frame. Additional time will need to be allotted for these activities unless you use them only for groups that finish the activities or tasks early.

CROSS-CURRICULUM CONNECTIONS

- The following cross-curriculum connections can be done as additional learning opportunities with the students and connect to different subjects.
    - History:
        - Work with students to research the history of robotics. Have them create a time line of notable events. You might want to search for "history for kids robotics" for some information or videos on the topic.
    - Math/Science:
        - Conduct a scavenger hunt for information on math or science topics currently being studied.
    - Arts:
        - View and evaluate a variety of dance performances. Discuss what makes dance amazing or boring. Help students identify different forms of dance.

ACTIVITY 1:

Scavenger Hunt

OUTCOME

- Use the TagTiles to become familiar with the map.

TIME

- 45 minutes

MATERIALS

- Task 1:
    - KUBO
    - Coding+ TagTiles
    - Movement TagTiles
    - Pencils
    - Small sticky notes (3 different colors)
    - Library activity map
- Task 2:
    - KUBO
    - Coding+ TagTiles
    - Movement TagTiles
    - Pencils
    - Library activity map
    - Small sticky notes (3 different colors)
    - Worksheet 3.1
- Task 3:
    - KUBO
    - Coding+ TagTiles
    - Movement TagTiles
    - Pencils
    - School activity map
    - Small sticky notes (3 different colors)
    - Worksheet 3.1

ACTIVITY 1:

Scavenger Hunt

TEACHER NOTES

In this activity, make sure to use small-size sticky notes so they fit inside the quadrants on the map.

Task 1:

- Before beginning, identify the different parts of the library on the map.
- KUBO is brand-new to the library, so KUBO's teacher has planned a scavenger hunt. A scavenger hunt is a game in which clues are given so the players can find different objects. In KUBO's game, there are three clues. Each clue will lead KUBO to a sticky note of a specific color.
- Explain what a scavenger hunt is if students are unfamiliar with it.
- Before starting, have students place a yellow sticky note on cell A4. Place a blue sticky note on cell E1. Place a pink sticky note on cell I6.
  - The color of the sticky notes can be modified. It's important that each sticky note be a different color so that you can easily check students' work.
- After each clue is provided, give students time to create their route to their destination. Constraints such as number of tiles, specific tiles, or time limit can increase the difficulty of this task.
  - Example: Create a route for KUBO in five minutes.
  - Example: Create a route for KUBO using only seven tiles. (This would encourage them to use the loop tile to reduce the tiles used.)
- When students have arrived at their destination, have them remove the sticky note where they arrived.
- Remind students to keep the sticky notes in the order they were obtained. After all three clues have been completed, have the students lay out their sticky notes in the order they were obtained. A quick walk around the room to view the sticky note order can determine if they went to each destination in the correct order.
- Clue 1 Answer: Travel/Geography section (A4: yellow sticky note)
- Clue 2 Answer: Biographies/People section (I6: pink sticky note)
- Clue 3 Answer: Animals section (E1: blue sticky note)

Task 2:

- Students create a scavenger hunt for another set of students. They need to write three clues where KUBO needs to go.
  - These destinations can reflect what students have been learning about during their library class, or they could be places the students choose.
- Students should place their three sticky notes on the map before giving the map and clues to another group.
- Sentence starters are just suggestions. Students should remember to make their clue specific enough that there is only one correct location.
  - Clues such as "KUBO loves to go here" are too vague and could be multiple places.
  - Encourage students to use facts to describe a location instead of opinions. Use something like "Looking at this will show you the world" for the globe instead of "This is a really cool tool to use."
- If there is time, students can switch clues and maps again.

ACTIVITY 1:

Scavenger Hunt

Task 3:

- Students will do the same activity as Task 2 but using the School activity map.

VOCABULARY

- Review Vocabulary
  - Coding+ TagTile: a tile in the KUBO Coding+ set; this set contains the following:
    - Direction TagTile: a tile that changes the direction KUBO is moving. Tiles include Turn 90 or 180 Degrees Left or Right, Go Backward, and Make a U-turn tiles.
    - Distance TagTile: a Movement tile that allows KUBO to move forward two, three, or four quadrants at a time
    - Speed TagTile: a tile that changes how fast KUBO moves for the remainder of the function; can make KUBO go slow, medium, or high speed
    - Time TagTile: a tile that makes KUBO pause for 2, 5, or 10 seconds at a time
- New Vocabulary
  - destination: a place you are heading toward
  - makerspace: an area where you can design, experiment, and explore with tools and materials to answer questions and problems in the world around you
  - plot: to plan
  - scavenger hunt: a game in which clues lead to a destination

DISCUSSION QUESTIONS

Task 1:

- Compare KUBO's library to the library at your school.
- Is there anything missing in the library that you would add?
- What kind of tiles did you use to get KUBO to each location?

Task 2:

- What did you do to make this a challenging task?
- What's an example of a clue you wrote?

Task 3:

- Did you prefer writing clues for the Library or School activity map?

REFLECTION

- Which tiles do you use frequently? Which tiles do you not use very often? How could KUBO use those tiles more often?
- What Coding+ tiles do you use the most? Are there any that you don't use very often? Why don't you use them often?

ACTIVITY 1:

Scavenger Hunt

EXTENSION

- Create a scavenger hunt for students in the classroom. The hunt can take place in your school's library, outside, or even throughout the whole school!
- Have students create their own maps for KUBO to use. Let students plan an environment for KUBO and then create a new scavenger hunt for KUBO. Switch with another team.

NOTES

ACTIVITY 2:

Multiple Subroutines

OUTCOME

- Create multiple subroutines embedded within other functions.

TIME

- 45 minutes

MATERIALS

- Task 1:
  - KUBO
  - Coding+ TagTiles
  - Movement TagTiles
  - Play and Record Function TagTiles
  - Library activity map
- Task 2:
  - KUBO
  - Coding+ TagTiles
  - Movement TagTiles
  - Play and Record Function TagTiles
  - Library activity map
- Task 3:
  - KUBO
  - Coding+ TagTiles
  - Movement TagTiles
  - Play and Record Function TagTiles
  - Library activity map

TEACHER NOTES

Task 1:

- When KUBO first comes into the library in the morning, certain jobs need to be completed.
- KUBO starts and ends ALL the jobs in the red square (call it home base), facing the door.
  - It is important that all jobs start and end in the same direction and location. This will enable students to mix and match the functions and place them in any order.
- Students will build a red function for KUBO to go to the lamp to turn it on and are instructed to have KUBO hurry.

ACTIVITY 2:

Multiple Subroutines

  - Students should use the Coding+ tiles for speed.

- Next, KUBO takes down the chairs at the table.

Task 2:

- Students will make a green function for KUBO to turn on the computers. After KUBO arrives at each computer, KUBO needs to wait a few seconds for the computer to turn on.
  - Students should use the Coding+ tiles for a delay at each computer.

Task 3:

- Students will embed three of the functions inside the fourth function. They will place the Play Function tile for the fourth function on home base and watch KUBO complete all the tasks. A list of all KUBO's tasks can also be called an algorithm. The embedded functions are subroutines.
  - Make sure KUBO memorizes the three functions inside the fourth function.

VOCABULARY

- Review Vocabulary
  - function: a route that can be memorized by KUBO with the use of the Record and Play Function TagTiles
  - Record and Play Function TagTiles: the tiles that enable KUBO to memorize a route as a function and then perform the function without following tiles
- New Vocabulary
  - algorithm: a recipe – step-by-step instructions to complete a task, usually used in computer programming and mathematics
  - embed: to place in the middle of something
  - task: something that needs to be done

DISCUSSION QUESTIONS

Task 1:

- What are some advantages of embedding functions?
  - Possible answer: Some TagTiles have only one of its kind in a set (for example, the Open Loop tile). By using an embedded function, KUBO can memorize one function, and then the tile can be reused in another function.
  - Possible answer: Functions can easily be rearranged in any order.

Task 2:

- What TagTiles did you use to ensure that KUBO waited at each computer for it to turn on?
  - Possible answer: a loop executed three times (one for each computer) with a delay in it
- Why is it important for KUBO to return to the same spot and face the same direction at the end of the code?
  - Possible answer: If KUBO ends the function facing a different direction or in a new location, the next function won't start where it should.

ACTIVITY 2:

Multiple Subroutines

Task 3:

- Did KUBO complete all four tasks correctly?
- What did you need to debug along the way?
- What would you need for KUBO to complete even more tasks?
  - Possible answer: If more Play and Record Function tiles were available in different colors, it would enable KUBO to complete more tasks.
- When would you use algorithms?
  - Possible answers: completing a long division problem, completing code for KUBO, writing computer code

REFLECTION

- Would you rather create multiple subroutines or create one long function for KUBO to execute? Why?
- Your friend wants to learn how to create multiple subroutines within a function. What tips would you give to help him or her succeed?

EXTENSION

- Using the other maps available for KUBO, have students create a list of tasks for KUBO to complete and then make the code to complete the tasks.
- Have a group of students predict what tasks another group wants KUBO to complete by looking at only their code and map.

ACTIVITY 2:

Multiple Subroutines

NOTES

ACTIVITY 3:

Pseudocode

OUTCOME

- Create pseudocode to plan more complex movements for KUBO.

TIME

- 45 minutes

MATERIALS

- Task 1:
  - KUBO
  - Coding+ TagTiles
  - Movement TagTiles
  - Play and Record Function TagTiles
  - Library activity map
  - Worksheet 3.3
  - Pencils
- Task 2:
  - Worksheet 3.3
  - Pencils
- Task 3:
  - Worksheet 3.3
  - Pencils

TEACHER NOTES

Task 1:

- When computer programmers create complex code, they often use pseudocode to plan their code first.
- Pseudocode is made of regular words to describe what you want to have happen. When your pseudocode is complete it breaks down each of the actions into smaller steps that can then be changed into code.
  - For example, if you want KUBO to go around an object such as the makerspace area in the library, the pseudocode would be to drive forward to the makerspace, turn left at the tables, turn right at the corner, do all that three times, and drive back to the door. Then you would use TagTiles to complete each of those actions.
- Students will write pseudocode for KUBO to enter the library, find a book on animals, and go to a table to read it.
  - Possible answer: Drive forward, go right around the tables, move forward to the animal bookshelf, turn around, and drive to a table.

ACTIVITY 3:

Pseudocode

- Students then select a new task for KUBO to complete and write the pseudocode for the task. Other tasks KUBO could do include:
    - Change the fish water.
    - Shelve books.
    - Go to the globe to check where Italy is found.
    - Select a book to learn about Albert Einstein and go to the makerspace area to read it.
    - Learn about the sinking of the Titanic by selecting a book on the topic and reading it at the computer tables.

Task 2:

- Students should brainstorm together the range of motions KUBO can complete and then share their ideas as a class. Some motions can include:
    - Go forward.
    - Go back.
    - Spin clockwise.
    - Spin counterclockwise.
    - Make a loop.
    - Move in a square.
    - Move in a rectangle.

Task 3:

- Students are instructed to KUBO is preparing for a dance competition. The competition has the following criteria:
    - The dance must be between 30-60 seconds.
    - The dance must have three main sections (functions).
    - Each section (function) should include two or three movements from KUBO's list of movements.
    - Students will be scored on creativity, variation of movement, and their pseudocode (whether they had a plan).
- Students are instructed to create pseudocode on their worksheet to explain what movements KUBO will do in the competition.
    - Students will need this worksheet for the next activity.
    - Decide if you are preselecting the music or if students can to select their own music.
    - KUBO can perform the dance on a map or on a table or floor.

VOCABULARY

- Review Vocabulary
    - function: a route that can be memorized by KUBO with the use of the Record and Play Function TagTiles
    - Record and Play Function TagTiles: the tiles that enable KUBO to memorize a route as a function and then perform the function without following tiles

ACTIVITY 3:

Pseudocode

- New Vocabulary
  - algorithm: a recipe – step-by-step instructions to complete a task, usually used in computer programming and mathematics
  - criteria: the limitations placed on a challenge
  - pseudocode: a list of actions that are translated into code

DISCUSSION QUESTIONS

Task 1:

- Did your pseudocode match what KUBO did?
- Did you need to change anything?

Task 2:

- What movement limitations does KUBO have?
  - Possible answers: KUBO spins in a circle or in right angles; KUBO doesn't have arms.

Task 3:

- How did you estimate how many dance steps you needed?
  - If students under- or overestimated, they can change their plan later.
- Why is pseudocode helpful?
  - Possible answer: Pseudocode helps you plan before you start coding. It ensures you have thought about all the steps you need.

REFLECTION

- Why do you think computer programmers use pseudocode?
- How can pseudocode help if you have a very complicated task?

EXTENSION

- Write pseudocode for a new task for KUBO to complete. Share with another group and have them use the TagTiles to complete the task based on your pseudocode.
- Make your dance more complicated by using loops or all of the Coding+ tiles.

ACTIVITY 3:

Pseudocode

NOTES

ACTIVITY 4:

Dance Off!

OUTCOME

- Create code for KUBO to compete in the dance competition.

TIME

- 45 minutes

MATERIALS

- Tasks 1-2:
  - KUBO
  - Coding+ TagTiles
  - Movement TagTiles
  - Play and Record Function TagTiles
  - Loop and Parameter TagTiles
  - Worksheet 3.3
  - Dance Scoring sheet
  - Pencils

TEACHER NOTES

Task 1:

- Using the pseudocode created in the previous lesson, students create code for KUBO to compete in the dance competition.
  - Recommended time limit for this task is 30 minutes. If students finish early, ask them how they can make KUBO's dance more challenging.
- If KUBO doesn't perform the way students want, have them make changes until KUBO has a winning performance.

Task 2:

- It's show time! Time to show off KUBO's dance moves! Students have KUBO perform the dance for their classmates.
  - Record or have students record KUBO's dance performance to grade later.
  - Students can grade each other's KUBO performances. This might help students stay engaged in the performances.
  - Consider setting up a video camera to a projector. This way students can watch the performance as it happens on the "big screen."

ACTIVITY 4:

Dance Off!

VOCABULARY

- Review Vocabulary
  - pseudocode: a list of actions that are translated into code
  - subroutine: a function inside another function

DISCUSSION QUESTIONS

Task 1:

- Were there any dance moves you planned for KUBO in the pseudocode that you couldn't use the TagTiles for?
- What were some problems you had with your code and TagTiles?

Task 2:

- When watching other KUBO performances, what did you like?
- What's a dance move you wish KUBO could do?

REFLECTION

- KUBO has done a lot of amazing things. What's something that KUBO has not done that KUBO should do next?
- Thinking about everything KUBO has done in all the lessons – what's your favorite task KUBO's done?

EXTENSION

- KUBO has completed a dance competition, but now KUBO wants a partner to dance with. Combine with another group and create a duet for two KUBOs will dance together. Remember, the KUBOs don't have to do the same movements.
  - Remind students to make sure KUBOs don't run into each other. Discuss dance steps that KUBO could do with the partner (for example, circle around each other, move toward each other, and then turn around).
- Sometimes dances incorporate props (items to dance with or around) and costumes to make a more engaging performance. Create props or costumes for KUBO. Make sure they do not interfere with KUBOs movement!

ACTIVITY 4:

Dance Off!

NOTES

Standards Addressed

US ISTE CURRICULUM STANDARDS

	KUBO CODING				KUBO CODING+		
Learning Outcome	**LP 1: Routes**	**LP 2: Functions**	**LP 3: Subroutines**	**LP 4: Loops**	**LP 1: Refresher course**	**LP 2: Advancing programming**	**LP 3: Challenge master**
1a Students articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.	●	●	●	●	●	●	●
1b Students build networks and customize their learning environments in ways that support the learning process.	●	●	●	●	●	●	●
1c Students use technology to seek feedback that informs and improves their practice and to demonstrate their learning in a variety of ways.	●	●	●	●	●	●	●
1d Students understand the fundamental concepts of technology operations, demonstrate the ability to choose, use and troubleshoot current technologies and are able to transfer their knowledge to explore emerging technologies.	●	●	●	●	●	●	●
2a Students cultivate and manage their digital identity and reputation and are aware of the permanence of their actions in the digital world.							
2b Students engage in positive, safe, legal, and ethical behavior when using technology, including social interactions online or when using networked devices.							
2c Students demonstrate an understanding of and respect for the rights and obligations of using and sharing intellectual property.							
2d Students manage their personal data to maintain digital privacy and security and are aware of data-collection technology used to track their navigation online.							
3a Students plan and employ effective research strategies to locate information and other resources for their intellectual or creative pursuits.							
3b Students evaluate the accuracy, perspective, credibility, and relevance of information, media, data, or other resources.							
3c Students curate information from digital resources using a variety of tools and methods to create collections of artifacts that demonstrate meaningful connections or conclusions.							
3d Students build knowledge by actively exploring real-world issues and problems, developing ideas and theories, and pursuing answers and solutions.					●	●	●
4a Students know and use a deliberate design process for generating ideas, testing theories, creating innovative artifacts or solving authentic problems.	●	●	●	●	●	●	●
4b Students select and use digital tools to plan and manage a design process that considers design constraints and calculated risks.	●	●	●	●	●	●	●

Standards Addressed

US ISTE CURRICULUM STANDARDS

	KUBO CODING				KUBO CODING+		
Learning Outcome	**LP 1: Routes**	**LP 2: Functions**	**LP 3: Subroutines**	**LP 4: Loops**	**LP 1: Refresher course**	**LP 2: Advancing programming**	**LP 3: Challenge master**
4c Students develop, test and refine prototypes as part of a cyclical design process.	●	●	●	●	●	●	●
4d Students exhibit a tolerance for ambiguity, perseverance and the capacity to work with open-ended problems.	●	●	●	●	●	●	●
5a Students formulate problem definitions suited for technology-assisted methods such as data analysis, abstract models and algorithmic thinking in exploring and finding solutions.	●	●	●	●	●	●	●
5b Students collect data or identify relevant data sets, use digital tools to analyze them, and represent data in various ways to facilitate problem-solving and decision-making.	●	●	●	●	●	●	●
5c Students break problems into component parts, extract key information, and develop descriptive models to understand complex systems or facilitate problem-solving.	●	●	●	●	●	●	●
5d Students understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions.	●	●	●	●	●	●	●
6a Students choose the appropriate platforms and tools for meeting the desired objectives of their creation or communication.	●	●	●	●	●	●	●
6b Students create original works or responsibly repurpose or remix digital resources into new creations.	●	●	●	●	●	●	●
6c Students communicate complex ideas clearly and effectively by creating or using a variety of digital objects such as visualizations, models or simulations.	●	●	●	●	●	●	●
6d Students publish or present content that customizes the message and medium for their intended audiences.	●	●	●	●	●	●	●
7a Students use digital tools to connect with learners from a variety of backgrounds and cultures, engaging with them in ways that broaden mutual understanding and learning.							
7b Students use collaborative technologies to work with others, including peers, experts, or community members, to examine issues and problems from multiple viewpoints.							
7c Students contribute constructively to project teams, assuming various roles and responsibilities to work effectively toward a common goal.					●	●	●
7d Students explore local and global issues and use collaborative technologies to work with others to investigate solutions.							●

Standards Addressed

UK NATIONAL CURRICULUM COMPUTER SCIENCE STANDARDS

			KUBO CODING				KUBO CODING+		
	Learning Outcome	**Curriculum Aspect**	**LP 1: Routes**	**LP 2: Functions**	**LP 3: Subroutines**	**LP 4: Loops**	**LP 1: Refresher course**	**LP 2: Advancing programming**	**LP 3: Challenge master**
AIMS	**The national curriculum for computing aims to ensure that all pupils:**								
	can understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation	CS	●	●	●	●	●	●	●
	can analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems	CS	●	●	●	●	●	●	●
	can evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems	IT	●	●	●	●	●	●	●
	are responsible, competent, confident and creative users of information and communication technology	DL	●	●	●	●	●	●	●
KEY STAGE 1	Understand what algorithms are	CS	●	●			●	●	●
	Understand that algorithms are implemented as programs on digital devices	CS	●	●				●	●
	Understand that programs execute by following precise and unambiguous instructions	CS	●	●			●	●	●
	Create simple programs	CS	●	●			●	●	●
	Debug simple programs	CS	●	●			●	●	●
	Use logical reasoning	CS	●	●			●	●	●
	Predict the behaviour of simple programs	CS	●	●			●	●	●
	Use technology purposefully to create, organise, store, manipulate and retrieve digital content	IT	●	●			●	●	●
	Recognise common uses of information technology beyond school	DL							

Standards Addressed

UK NATIONAL CURRICULUM COMPUTER SCIENCE STANDARDS

		KUBO CODING					KUBO CODING+		
	Learning Outcome	**Curriculum Aspect**	**LP 1: Routes**	**LP 2: Functions**	**LP 3: Subroutines**	**LP 4: Loops**	**LP 1: Refresher course**	**LP 2: Advancing programming**	**LP 3: Challenge master**
KEY STAGE 1	Use technology safely and respectfully	**DL**	●	●			●	●	●
	Keep personal information private	**DL**							
	Identify where to go for help and support when they have concerns about content or contact on the internet or other online technologies.	**DL**							
KEY STAGE 2	Design programs that accomplish specific goals	**CS**	●	●	●	●	●	●	●
	Write programs that accomplish specific goals	**CS**	●	●	●	●	●	●	●
	Debug programs that accomplish specific goals	**CS**	●	●	●	●	●	●	●
	Control or simulate physical systems	**CS**	●	●	●	●	●	●	●
	Solve problems by decomposing them into smaller parts	**CS**			●	●	●	●	●
	Use sequence in programs	**CS**	●	●	●	●	●	●	●
	Use selection in programs	**CS**							
	Use repetition in programs	**CS**				●	●	●	●
	Work with variables	**CS**							
	Work with inputs	**CS**	●	●	●	●	●	●	●

Standards Addressed

UK NATIONAL CURRICULUM COMPUTER SCIENCE STANDARDS

			KUBO CODING				KUBO CODING+		
	Learning Outcome	**Curriculum Aspect**	**LP 1: Routes**	**LP 2: Functions**	**LP 3: Subroutines**	**LP 4: Loops**	**LP 1: Refresher course**	**LP 2: Advancing programming**	**LP 3: Challenge master**
KEY STAGE 2	Work with outputs	**CS**	●	●	●	●	●	●	●
	Use logical reasoning to explain how some simple algorithms work	**CS**	●	●	●	●	●	●	●
	Use logical reasoning to detect and correct errors in algorithms and programs	**CS**	●	●	●	●	●	●	●
	Understand computer networks including the internet	**CS**							
	Understand they can provide multiple services, such as the world wide web	**CS**							
	Understand the opportunities they offer for communication and collaboration	**DL**							
	Use search technologies effectively	**IT**							
	Appreciate how results are selected and ranked	**CS**	●	●	●	●	●	●	●
	Be descerning in evaluating digital content	**DL**							
	Select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information	**IT**							
	Use technology safely, respectfully and responsibly	**DL**	●	●	●	●	●	●	●
	Recognise acceptable/unacceptable behaviour	**DL**							
	Identify a range of ways to report concerns about content and contact	**DL**							