



# Quick start guide

KUBO CODING++



**KUBO is the world's first puzzle-based educational robot, designed to empower students so that they are not just passive consumers of technology, but rather controllers and creators of technology. By simplifying complex concepts through hands-on experiences, KUBO builds confidence among educators and students and provides a context for endless possibilities to engage in playful STEAM activities.**

**KUBO and the unique TagTile® programming language lays the foundations for computational literacy for children aged four to 10+.**



## Getting Started

This Quick Start Guide explains what is included in your KUBO Coding++ TagTile® Set and provides examples of how to expand your students' programming skills.

To use this set, you will need a KUBO Coding Set.

### WHAT'S IN THE BOX

Your KUBO Coding++ TagTile Set includes 44 TagTiles stored in a convenient sorting tray. The set allows you to introduce concepts of variables, conditions (if/else true or false) and events. Read on for a full description of all the tiles.



# KUBO Coding++ TagTiles®



The KUBO Coding++ TagTile® Set is an add-on set, which completes the introduction of the essentials of coding with KUBO. With KUBO Coding++ you can continue to develop your students coding skills in the same screen-free, hands-on way, so that they gain the confidence to experiment and explore the endless possibilities that coding offers. The lesson plans included in the set meet most ISTE and CSTA Computer Science standards for students aged 7 to 10+ (K3-K5).



In your KUBO Coding++ TagTile® set you will see three sections:

## Section **1** TagTiles®

### VARIABLES

Variable TagTiles are used for storing a number/parameter. A parameter can be used to modify or change the value of a variable, hence the name variable. There is almost no limit to the number the variable can contain. Variables don't have to be set within a function and can be used outside of a function, for example in a simple route.

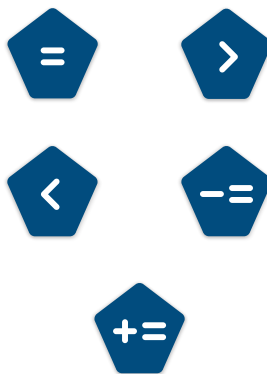
There are three blue and three orange Variable TagTiles.



## OPERATORS

Operator TagTiles® are used to change or check the value of a variable. When checking, the user can use =, >, < to see if the variable is equal to, more than or less than a value. If you want to increase or decrease the value of the variable, you use the += or -= tiles.

A negative value in a loop will result in no execution of the loop. A negative value in set speed and angle tiles will make KUBO act the opposite of its positive value. There are eight operator tiles; two Add to Variable; two Subtract from Variable; two Equal; one Greater Than and one Less Than TagTile.



## MODULATORS

The Modulator TagTile is used when changing or checking the value of a variable. It provides a way to help students put the right tiles in the right order to change or check a variable against a given value. This is typically done to program a condition. There are three modulator tiles included in the set.

## ZERO PARAMETER

The Zero Parameter TagTile is an addition to the 1-10 parameter tiles that you already have in your KUBO Coding Set. This parameter can be used in the same way as the others, for example in loops, to set speed, control angles and turns or to manipulate variables. There is one Zero Parameter tile.



# Section **2** TagTiles®

## CONDITIONS

The If/else, True and False Condition TagTiles® allow you to define a sequence of behaviors depending on whether a condition is true or false. There are eight Condition TagTiles, two Start If and two End If, two True and two False tiles.



**Start If and End If TagTiles**



**True and False TagTiles**



## EVENTS

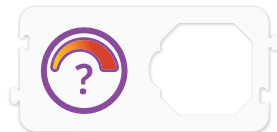
Event TagTiles are used together with Condition TagTiles and allow you to define an alternative sequence of behaviour for KUBO to execute, if the robot meets an event tile on the activity map. Students can draw on the blank event tiles to make them specific to their story. For example, the tile is illustrated with a monster, if KUBO meets the monster he should make a u-turn and move forward x3 at high speed.

The event tiles are reusable providing students use a dry wipe marker pen. There are three sets of event tiles, in blue, red and orange. Each set includes one Event Code tile to place into the code and three Event Map tiles to place on an activity map.

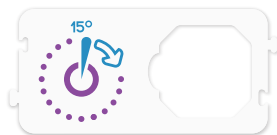
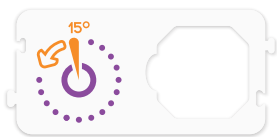
# Section **3** TagTiles®

## SET SPEED WITH PARAMETER

This TagTile® allows you to control the speed that KUBO travels. KUBO operates on a 10-scale where 1 is very slow and 10 is fast. Parameter Zero (0) will make KUBO stand still. There are two possible speed settings for KUBO; one that is set outside of a function and one within the function. You cannot use both settings at the same time because KUBO will not remember a speed setting when executing a route and then a function, or vice-versa.



There are two Speed with Parameter TagTiles.

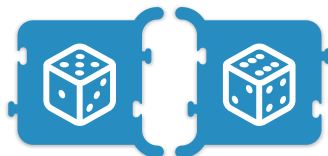


## TURN 15 DEGREES LEFT OR RIGHT WITH PARAMETER

With these TagTiles KUBO will turn the number of degrees to the left or the right and will repeat that action the number of times stipulated in the parameter. For example place the number 4 in the space and KUBO will turn 4 x 15 degrees. You can also use a variable to determine how many degrees KUBO should turn. You can only use these tiles in a function where you can add a tile to tell KUBO what the next move after the turn should be. NOTE: If you use a negative value in these tiles, KUBO will execute the value in reverse. There is one 15 Degree Turn Right and one 15 Degree Turn Left tile.

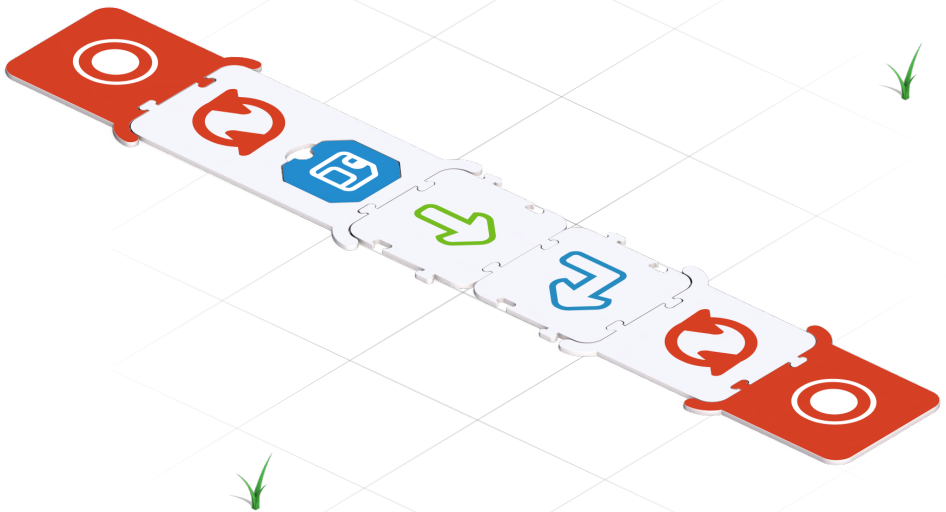
## START/END RANDOM

The Random TagTiles are used in a function to make KUBO execute commands in a random order and number. There are two random tiles.



# Sample Programs using KUBO Coding++

The following examples demonstrate how to use  
KUBO Coding++ TagTiles® when programming  
variables, conditions, events and random sequences.





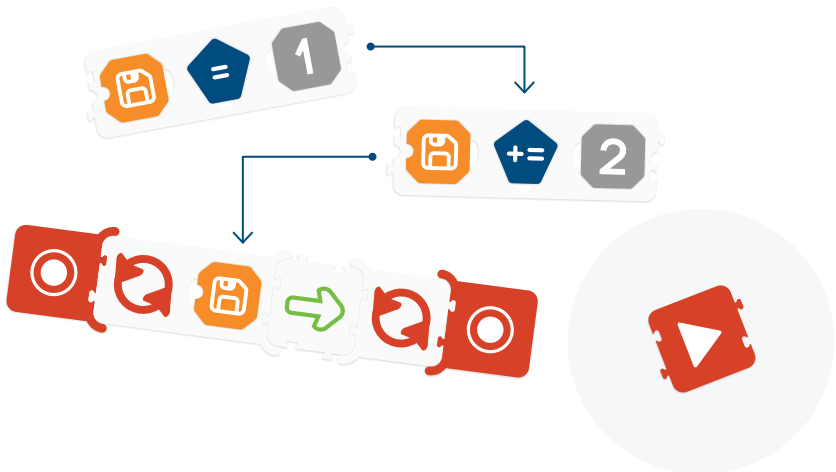
# Programming with Variables

By making use of **variables** in programs, students learn how data is stored and can be manipulated by the programmer. In terms of cross-curricular activities, variables can also be used to represent many different elements that have a value, for example not just numbers, but also liquids, metals, letters and so on.

In the following example using numbers, the orange variable on the top row is set to a value equal to one. The second variable is set to a value of plus two.

This means that the total value of the variable will be  $1 + 2 = 3$

The function with the loop tells KUBO that the variable is 3, and therefore KUBO must Go Forward x3.



## NOTE:

(I) The modulator can be used outside of a function, to set or modify a variable as shown above. When used inside a function, it is used to check a condition.

(II) When you place KUBO on the variable outside of a function it will beep the value that the variable has been given. As this number can be in the 1,000s KUBO will count in the following way: First KUBO moves forward, turns and counts the number of thousands, with one beep per thousand, then moves forward one. Then KUBO counts the number of hundreds, with one beep per hundred and moves forward one. Then KUBO counts the number of tens with one beep per ten, and moves forward one. And finally KUBO counts the number of single digits with one beep per digit, without moving forward. KUBO will remember the most recent variable value even after its head has been removed.

# Programming with Conditions

When making use of **conditions** children experience first hand how cause and effect can be controlled; If x is true, do y, if not, do z. In coding terms this is often referred to as “if/else”.

Programming with these TagTiles® will allow KUBO to execute a specific sequence of commands if a specific condition is true, or to execute an alternative sequence if the condition is false. Use the modulator to check if something is true or false, or use the Event TagTiles to control alternative sequences for KUBO to execute on an activity map.

In the following example the blue Variable is set to the value of three.

The Function tells KUBO: “If the blue Variable is more than 6 then Go Forward x1, else Go Left.” In this example KUBO will Go Left.



A Condition TagTile can be used with or without a True or False TagTile, i.e it does not need to include both. However, to understand the principle of Conditions it is better to use both.

## NOTE:

Remember to use conditions within a function and to execute using the corresponding Play Function TagTile.

# Programming Events

Event TagTiles® allow KUBO to react to the environment, which makes coding activities even more playful and dynamic.

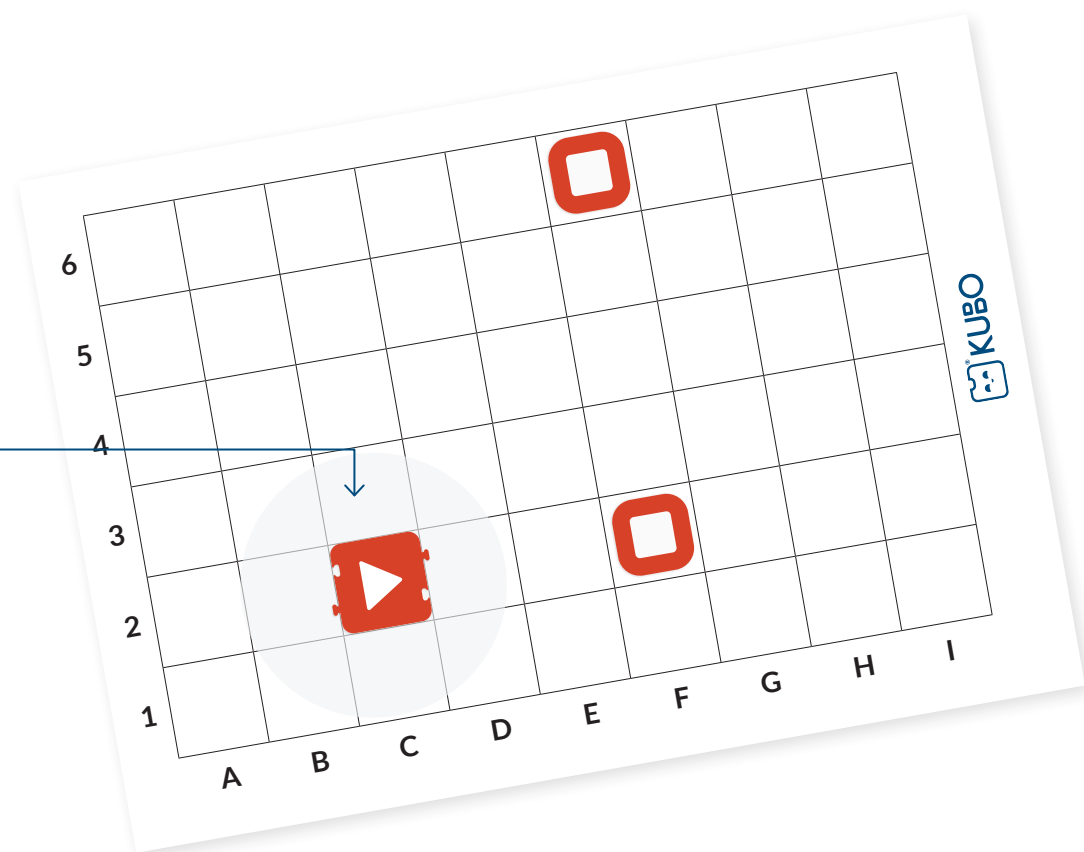
The Event Code TagTile used within a function will define what KUBO needs to react to. The smaller Event Map TagTiles in the same color, are used on an activity map. These will define the “True” sequence of commands you decide upon. If KUBO does not meet an Event Map TagTile, the robot will execute the “False” or alternative sequence.

In the following example, the function determines that “If KUBO meets an Event Map TagTile, *then* turn left, *else* (meaning, if KUBO doesn’t meet an event tile) go forward x1. This example includes a recursive function (the red Play Function TagTile inside the Function), which means that KUBO will keep going forward x1 until the robot meets an Event Map TagTile.



For more ideas and support go to [www.kubo.education](http://www.kubo.education)

There are free lesson plans and activities that challenge students to improve their programming skills using KUBO Coding++ TagTiles. You can also watch short video tutorials on the website.



**NOTE:**  
Event TagTiles® can only be  
used with a condition.

# Programming random sequences

The concept of random is more mathematical than coding, although random is used frequently in coding. Using the Random TagTiles® within a function you can place as many tiles as you want between the random tiles, but only a random selection of the tiles will be executed by KUBO. This adds a playful element to coding and facilitates the learning of important math concepts such as probability, fractions and percentages.

In this example KUBO will choose a random sequence consisting of one or more commands; Go Left, Go Forward x1 or Go Right.



**NOTE:**

Remember always to use Random TagTiles inside a function. You can also only use Movement TagTiles inside a random function.



For more ideas and support go to [www.kubo.education](http://www.kubo.education)

There are free lesson plans and activities that challenge students to improve their programming skills using KUBO Coding++ TagTiles. You can also watch short video tutorials on the website.

# The Coding License lesson plans

The Coding License, available to view or download at [www.kubo.education](http://www.kubo.education), provides a comprehensive set of lesson plans and teacher guides designed to take teachers and students through every KUBO concept of coding in a playful, progressive and creative way.

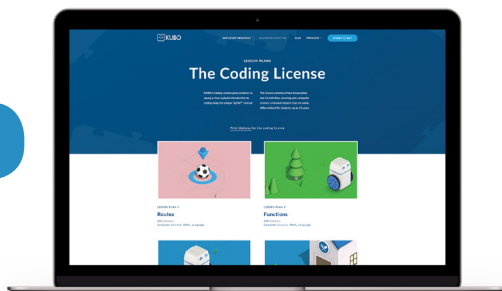
## MATERIALS INCLUDE:

- 4 lesson plans, with 16 25-minute activities introducing coding using the KUBO Coding Set.
- 3 lesson plans, with 12 45-minute activities introducing coding using the KUBO Coding+ TagTile Set.
- 1 lesson plan, with three 45 to 90-minute activities introducing the KUBO Coding++ TagTile® Set, plus a project pack for activities that can run over several days or weeks.

## EACH LESSON PLAN INCLUDES:

- Appealing and playful 3D illustrations that students will love
- Story-starters to stimulate their motivation and engagement
- Worksheets to download
- Task cards to encourage self-directed learning
- A Coding License certificate that tracks progress
- Assessment and extension ideas
- A printable Teacher's Guide

[WWW.KUBO.EDUCATION](http://WWW.KUBO.EDUCATION)



# KUBO Coding++ progression



Students can start KUBO at any grade from K-5, and will continuously develop their coding skills at a pace determined by the teacher.



All rights reserved © 2019  
KUBO Robotics ApS  
Niels Bohrs Allé 185 5220 Odense SØ  
SE/CVR-nr.: 37043858

[www.kubo.education](http://www.kubo.education)