

Hint

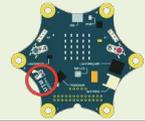
Don't forget:
The shaking cannot be tested, therefore download and transfer.

Info

Coincidence is actually something a computer doesn't know. It always does exactly what you program. The random number block must therefore be used consciously.

Start

- Enter lab.open-roberta.org into a browser. → This is where you can start programming.
- As far as the system goes choose Calliope and then Calliope 2017.



Task

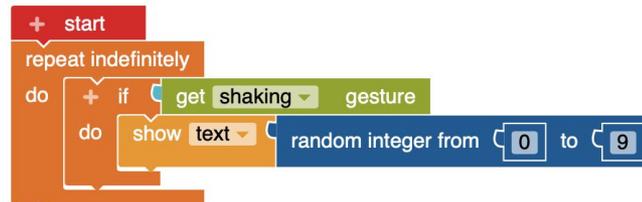
We'll program a dice that shows a random number between 0 and 9 if shaken. You first need the "repeat indefinitely... do" and then the "if... do" block.

1. Drag **get upright gesture** from **Sensors** to the programming environment. Press the small arrow and select *shaken*.

2. Add **Show Text** delete the "Hello"-block.

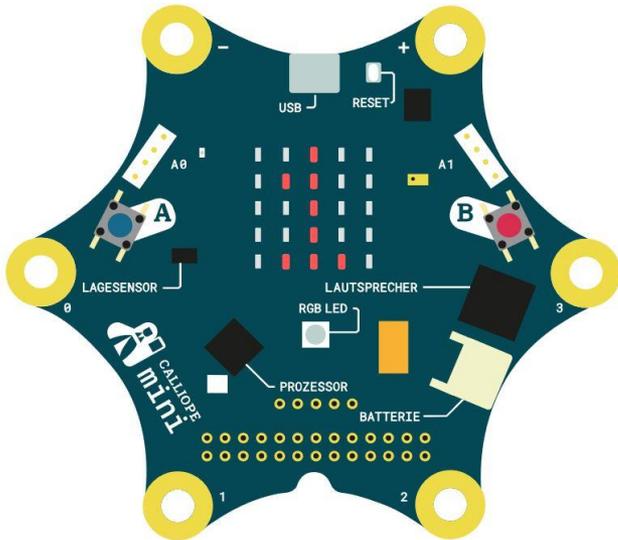


3. In the advanced functions search for the block Math which can create random values.
4. Adjust the block so that a number between 0 and 9 is selected.



Try it out!

Find another person or group who has already solved this task. Now you can play:
Which Calliope rolls the higher number?
If everything goes well, get the next flashcard.



Hint

1. Click on the small plus sign in the corner when starting.
2. Rename the variable name (element) to random.



Info

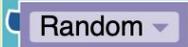
Variables work like a container. For example, you can insert a number and retrieve it later. So the Calliope mini can "remember" things.

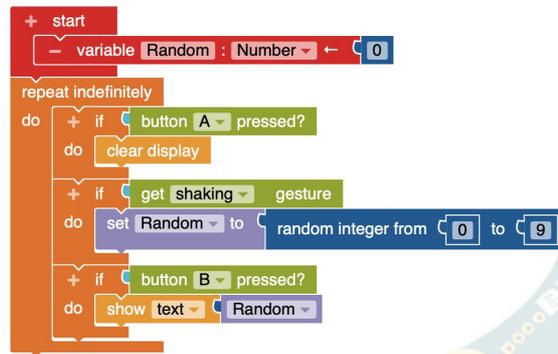
Start

- **Keep** the blocks of the previous flashcard.

Task

We change the cube so that the Calliope mini remembers a random number while shaking. However, the number is only displayed when button B is pressed.

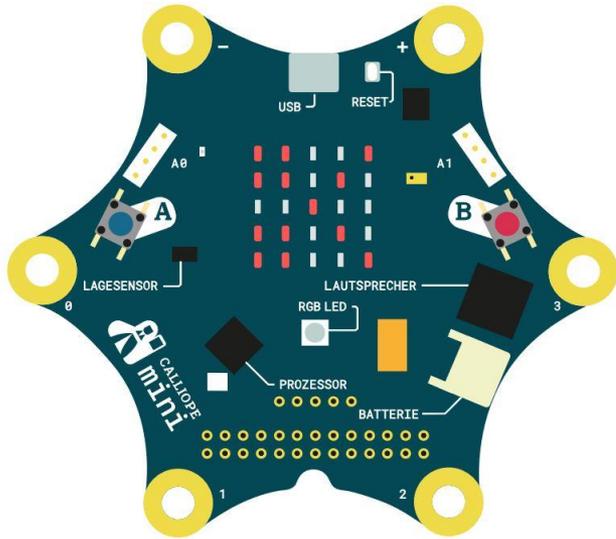
1. Create a new variable and call it random. On the left you can see how it works.
2. Bei **"Lage geschüttelt aktiv"**:
- Use the block **"Write random"**, which you can find under variables.
3. Add **"integer random value"** to the **"write random"** block.
4. **"If button B pressed"**, **"Zeige text"** 



Try it out!

Test your program with another person. Does everything work?

In the next step we'll program scissors, stone, paper.
Get the next flashcard.



Task

Delete the blocks of the previous flashcard.

1. Determine a new **variable** Random and set it to -1 at the beginning so that no scissors are displayed in the beginning.
2. **If button A is pressed**, the screen should be cleared.
3. Whenever "**get upright gesture**" is active, a random value between 0 and 2 is to be stored in the Random variable.
4. Under **logic** you will find . This allows you to check which number is currently stored in your random variable.:

- If Random is equal to 0, then Show Image Scissors.
- Else if coincidence equals 1, then show picture stone.
- Else if Random is equal to 2, then Show picture paper.

```

+ start
- variable Random : Number +- 0
repeat indefinitely
+ - if button A pressed?
do
clear display
else if get shaking gesture
do
set Random to random integer from 0 to 2
+ - if Random == 0
do
show image 0 0 1 2 3 4
0 # # # #
1 # # # #
2 # # # #
3 # # # #
4 # # # #
else if Random == 1
do
show image 0 0 1 2 3 4
0 # # # #
1 # # # #
2 # # # #
3 # # # #
4 # # # #
else if Random == 2
do
show image 0 0 1 2 3 4
0 # # # #
1 # # # #
2 # # # #
3 # # # #
4 # # # #
    
```

```

+ - if Random == 0
do
else if Random == 1
do
else if Random == 2
do
    
```

Hint

Don't forget to add
after the start.

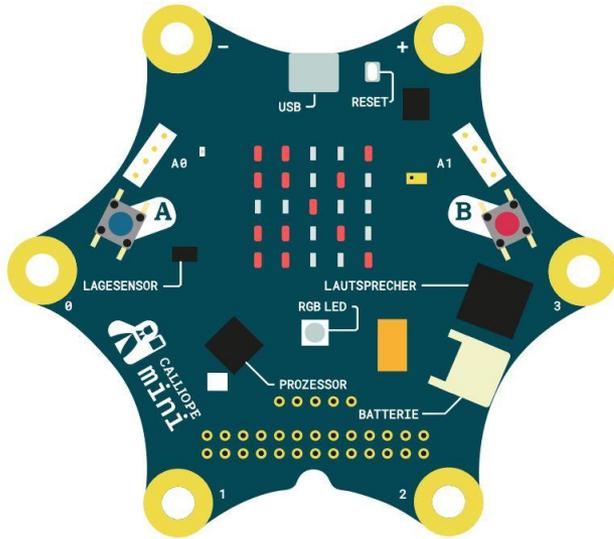


Info

The **if - do** block is very important when programming. It teaches a computer to think logically. It always works according to the same principle, for example:
If you are at home, **do** your homework.

Try it out!

Play against your Calliope mini or let two play against each other. After the next flash card, the Calliope mini can count your points.



Info

Variables are called that because they are variable, i.e. they can be changed. In this example we change the value of the variable My_Points whenever the player gets a point. So the variable changes the value, first 0, then 1, then 2 and so on.

Task

Now we are expanding the game so that the Calliope mini counts how many points you already have at the touch of a button. So keep the blocks from the previous flashcard.

1. Create a new variable and name it My_Points.
2. If you click on key B, My_Points is to be increased by 1 point.
3. Show your current score My_Points, if Layer upside down is active.

```

+ start
- variable Random : Number ← -1
- variable MyPoints : Number ← 0

repeat indefinitely
do
+ if button A pressed?
do
clear display
else if get shaking gesture
do
set Random to random integer from 0 to 2
+ if Random == 0
do
show image 0 1 2 3 4
1 # # # #
2 # # # #
3 # # # #
4 # # # #
else if Random == 1
do
show image 0 1 2 3 4
0 # # # #
1 # # # #
2 # # # #
3 # # # #
4 # # # #
else if Random == 2
do
show image 0 1 2 3 4
0 # # # #
1 # # # #
2 # # # #
3 # # # #
4 # # # #
else if button B pressed?
do
change MyPoints by 1
+ wait until get pressed button B == false
else if get upside down gesture
do
show text MyPoints
    
```

Info

After increasing the points, wait until key B is released again (press key B = wrong), otherwise the score will not be increased by 1, but by 30 or 40, because the Calliope mini counts internally very fast.

Done already?

You got to know the random function today.

Now try for yourself what is possible. At the end of the lesson you can show what you have accomplished in your class.

You could, for example:

- program a normal cube which shows the pictures 1-6 with the LEDs.
- program a machine that splits anyone who shakes into a team.
- to program a "truth or dare" game.
- ...

