

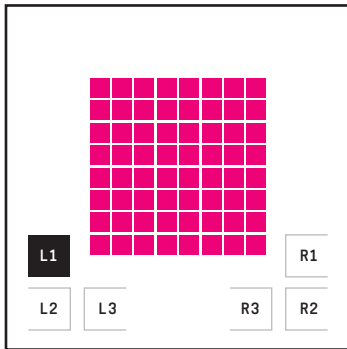
WORKSHEET W019

LIST

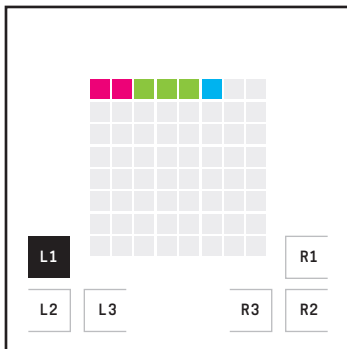
In this worksheet, we'll look at how the color list works.

EXERCISE

When a button on the left is pressed (L1, L2 or L3), a different color should be displayed briefly and then added to the color list. If you press a right button (e.g. R1), the stored sequence will be displayed as single pixels.



L1, for example, lets the LED matrix light up red briefly and stores the red value in the color list. For example, L2 could be green and L3 blue.

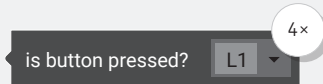
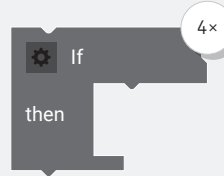


R1 then displays the saved list as individual pixels. In this example, 2xL1, 3xL2, and 1xL3 were pressed.

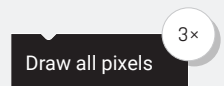
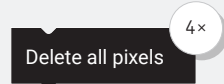
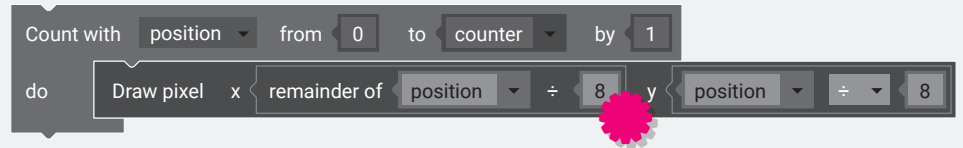
HINT

Use the template available in the parts list to draw the pixels in the loop. All you have to do is read the color from the list and set it.

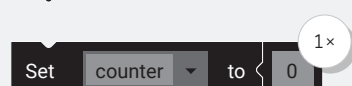
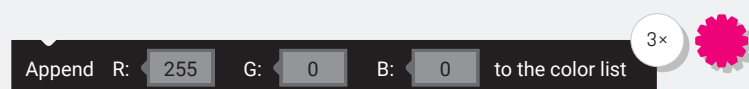
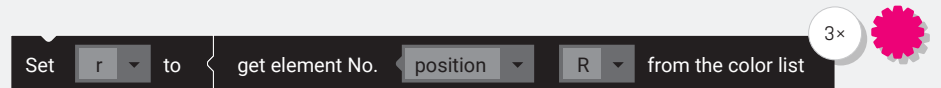
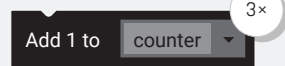
LOGIC



MATRIX



VARIABLES



TIME



PARTS LIST LIST OF BLOCKS TO BE USED



LEVEL EXPERT

SOLUTION

PROPOSED SOLUTION

The code starts with a 'Set counter to 0' block. It then enters a 'Do forever' loop containing four 'if' blocks for buttons L1, L2, L3, and R1. Each 'if' block has a 'then' section with the following steps: 'Append' a color (R, G, or B) to the color list, 'Add 1 to counter', 'Set pixel color' to the corresponding color, 'Draw all pixels', 'Wait 500 milliseconds', and 'Delete all pixels'. The R1 block includes a 'Count with' loop from 0 to counter by 1, with a 'do' section that sets variables r, g, and b to the corresponding elements in the color list, sets the pixel color to r, g, and b, draws the pixel at a grid position (x = remainder of position / 8, y = position / 8), waits 50 milliseconds, and then waits 3000 milliseconds before deleting all pixels.

```
Set counter to 0

Do forever
  If is button pressed? L1
  then
    Append R: 255 G: 0 B: 0 to the color list
    Add 1 to counter
    Set pixel color 255 0 0
    Draw all pixels
    Wait 500 milliseconds
    Delete all pixels

  If is button pressed? L2
  then
    Append R: 0 G: 255 B: 0 to the color list
    Add 1 to counter
    Set pixel color 0 255 0
    Draw all pixels
    Wait 500 milliseconds
    Delete all pixels

  If is button pressed? L3
  then
    Append R: 0 G: 0 B: 255 to the color list
    Add 1 to counter
    Set pixel color 0 0 255
    Draw all pixels
    Wait 500 milliseconds
    Delete all pixels

  If is button pressed? R1
  then
    Count with position from 0 to counter by 1
    do
      Set r to get element No. position R from the color list
      Set g to get element No. position G from the color list
      Set b to get element No. position B from the color list
      Set pixel color r g b
      Draw pixel x remainder of position / 8 y position / 8
      Wait 50 milliseconds
    end
    Wait 3000 milliseconds
    Delete all pixels
```

WORKSHEET W019

Learning objective:

Capacity to know how lists work.

WHAT TO DO

1. First you set the counter variable to zero.
2. Then the «Do forever» loop is needed. All other blocks must be positioned in this loop.
3. In this loop are all «if/then» blocks, which all query another button. In the first three «then» columns a different color is added to the color list, the counter variable is increased by one and the respective color is displayed for a short time.
4. In the last «then» column, all elements in the color list are to be read and output in a loop. Since each color element has three values (R, G, B), three individual variables (r, g, b) must be set. Now set the pixel color and draw the pixel at the corresponding position. For a great filling effect you can insert a short «Wait» block in the loop.
5. Outside the «Count with position from 0 to ,counter' in steps of 1» loop, another «Wait» block is needed so that the drawn pixels can be seen for a while. Finally, the pixels can optionally be deleted.



This is an «Expert-Block» and is only displayed if «Settings» – «Activate the Expert-Mode» is set.

Click on «Settings» in the lower left corner ...



Settings

... and select «Activate the Expert-Mode».

NEW COMMANDS

Append R: 255 G: 0 B: 0 to the color list

With the «Append RGB to color list» block any color can be added to a predefined list. Each further call of this block adds a new element to the list.

get element No. 0 R from the color list

A previously added element of the color list can be read again with the «get element no. x from the color list» block. Since a color has three values, the block must be called three times to assign all colors to a variable, for example.

Like almost everything in programming, the lists are zero indexed. So the first element of the list has the number 0.

ADDITIONAL INFORMATION: LISTS

The advantage of lists of conventional variables is that almost any number of elements can be added and removed at runtime. Lists are also a practical tool for games. The state of several game elements can be saved and retrieved later without having to declare a variable for everything. For example, if a sequence needs to be saved or a queue needs to be implemented in a game.

