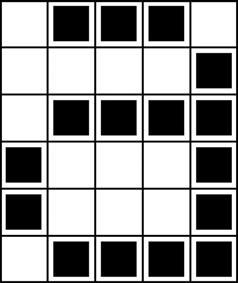
**Image Representation Worksheet** ([Click here to copy](https://docs.google.com/document/d/1AkIwOQLTU4_TonpRh3LEqoLMXWiVdZ4AiYf1y-qWIEI/copy))

*This handout is adapted from the* [*CS Unplugged Image Representation Activity*](https://classic.csunplugged.org/wp-content/uploads/2014/12/unplugged-02-image_representation.pdf)  
**Introduction**  
How can computers store pictures when they can only use binary numbers? How can we compress images so that they are faster to transmit and easier to store? Screens and images are divided up into a grid of small dots called **pixels** (picture elements). The letter “a” has been magnified below to show the pixels. When a computer stores a picture, all that it needs to store is which dots are black and which are white.



1, 3, 1

4, 1

1, 4

0, 1, 3 ,1

0, 1, 3, 1

1, 4

The picture above shows us how a picture can be represented by numbers using **Run Length Encoding** (RLE) used in fax machines. The first line consists of one white pixel, then three black, then one white pixel, represented as 1, 3, 1. The first number always relates to the number of white pixels. If the first pixel is black, the line will begin with a zero.

In your POGIL groups or in pairs, do the following activities on paper or using the interactive grid on the lesson page.

1. Try drawing in the following RLE compression and see if you get something you recognize. Remember the first number in each row is the number of white pixels.  
    0, 6  
    4, 1, 1  
    3, 1, 2  
    2, 1, 3  
    1, 1, 4  
    0, 6

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | 1 | 1 | 1 | 1 | 1 |
|  |  |  |  | 1 |  |
|  |  |  | 1 |  |  |
|  |  | 1 |  |  |  |
|  | 1 |  |  |  |  |
| 1 | 1 | 1 | 1 | 1 | 1 |

2. Each student should create their own black and white picture and then write down the RLE encoding for it.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 | 1 |

0,3,1,2

0,2,1,3

0,1,1,4

0,1,1,4

0,2,1,3

0,3,1,2

Fill in the RLE numbers for each row below (remember to start with the number of white pixels (0 if there are none) and alternate) and then tear or fold here and give to your partner to see if they can decode it and draw your image.

--------------------------------------------------------------------------------------------------------------------------------

**Run-Length Encoding:**

**Row 1: 6**

**Row 2: 0,6**

**Row 3: 6**

**Row 4: 0,6**

**Row 5: 6**

**Row 6: 0,6**

When you receive the RLE numbers above, draw the represented image (remember to start with the number of white pixels and alternate).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

3. **Color Images:** Using colored pencils or a shading pattern, draw an image with 3 different colors.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Fill in the color codes and the RLE numbers for each row below. Use an initial number to represent the color (for example 0 for white, 1 for black, 2 for red or diagonally shaded) before you write down the number of pixels of that color. For example, 0, 4, 2, 3 means white 4 pixels followed by red 3 pixels in that row. Tear or fold here and give to your partner to see if they can decode it and draw your image.

-------------------------------------------------------------------------------------------------------------------------------

**Run-Length Encoding:**

**Color Key: 0 = Orange**

**1 = Black**

**2 = Red**

**Row 1: 0, 4, 1, 0, 2, 2**

**Row 2: 0, 0, 1, 4, 2, 2**

**Row 3: 0, 4, 1, 0, 2, 2**

**Row 4: 0, 0, 1, 4, 2, 2**

**Row 5: 0, 4, 1, 0, 2, 2**

**Row 6: 0, 0, 1, 4, 2, 2**

When you receive the RLE numbers above, draw the represented image.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |