Run-Length Encoding - Reading

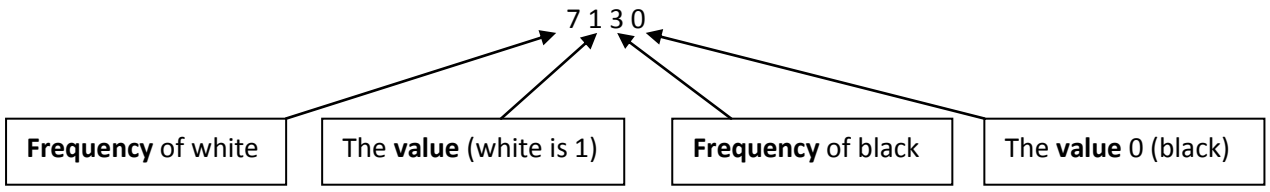
One of the easiest ways to **compress** a file is known as r**un-length encoding (RLE)**. This works well in files where there are large amounts of data with the same value.

This is very useful in images as the previous and next colour will often be the same. Instead of storing what the value for each pixel is, we instead store how many of each colour there are in sequence.

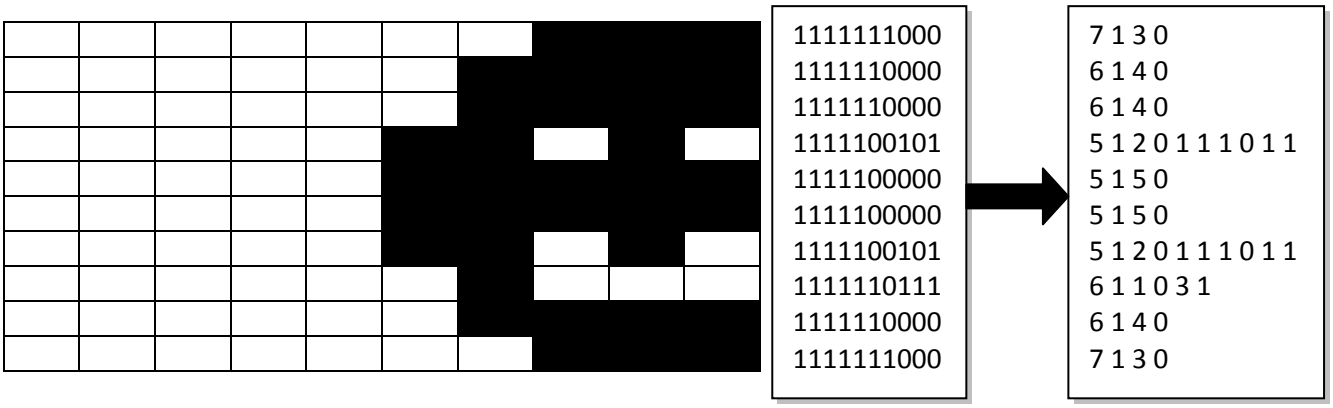
In the following example, for the first row, there are 7 white pixels followed by 3 black pixels. This would be stored **uncompressed** as:

1111111000

We now compress the data by placing the **frequency** of one colour followed by the **value** of that colour. So the first row would become:



Original bitmap image **Binary data** **RLE data**



We want to be able to compress data like this so that it requires less **memory**. This will result in it taking up less **storage** space on a **disk** or in **RAM**. It will also be **transmitted** faster through a **network** such as the **Internet**.

This technique is simple to understand and for a computer to **process**. It was therefore popular when computers had slower processors. More complicated **algorithms** are now used for most **image compression**. However, this method is very suitable for compressing fax images, which usually contain text and therefore large areas of white.

Run-length encoding is a lossless method of compression. This means that we can uncompress RLE data back to exactly the original data we compressed.