STRINGS OF DATA

The Square Kilometre Array is going to collect mind-boggling amounts of information. Information needs to be stored and displayed accurately to be truly useful.

One of the most amazing things that the *SKA* will have to deal with is the sheer amount of information it will generate. Unlike optical telescopes, radio telescopes don't actually record images. They record the strength of signals within a given stretch of the electromagnetic spectrum. The strength of that signal is converted to a string of numbers. That string is what gets stored, and used to make pictures.

This system is also how a digital camera works- though digital cameras detect visible light instead of radio waves.

This activity is going to explore the idea of representing pictures (in this case *text*) as pixels- small squares that make up a picture. The colour and position of that pixel is determined by the string of numbers.

To keep it relatively simple, in this activity, we will use 1 to mean "fill the square" (you can either fill it completely, or mark it ☒) and 0 to mean "leave it empty". With that rule established, we can create strings, or read them.

What to do

Reading a string of data

The first thing to figure out with a string is the size of the rectangular grid that it fits into.

As an example, the string (111101111100100) is bristling with information.

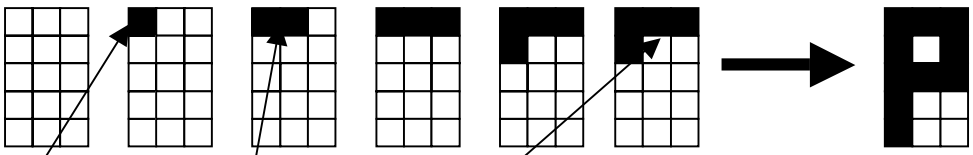
It's 15 numbers long. We have to fit this string into a grid, so we can look at the factors of 15 to tell us how big the grid is going to be.

The factor pairs of 15 are (1,15) and (3,5). From here, you can decide the dimensions of the grid.

We've constructed each grid to have prime sides. Accordingly, when you count how long the string is, looking at the factors of that number will give you the grid dimensions.

If you have a string that has 121 entries, then it's going to be either 1×121 or 11×11 grid. (Why can we ignore the 1×121 grid?)

A string of 77 would form a 7×11 grid.



111101111100100
This 1 tells us to fill the first square. Move to the next part of the string…

1**1**1101111100100
This 1 tells us to fill the second square, and so on…

1111**0**1111100100
This 0 tells us to leave the fifth square empty….

The longer the string, the more information there is in it. This becomes really important when you start thinking about the *resolution* of the picture. If you draw a single letter in a 11×37 grid, the level of detail (for example how smooth the curves are) can be much greater than if you represent the same letter in a 3×7 rectangular grid.

Examples

Here are some example strings- can you find out what they say?

1:

0010001010100011000111111000110001

2:

1111010010100101111110001100011111

3:

0111010001100001000010000100010111 0

4:

1110010010100011000110001100101110 0

5:

1111110000100001110010000100001111 1

6:

1000110001100011010110101101010101 0

7:

0111001000100010000100010100100010100010000010100010001000111001100
00100010000001010100011111001000101001001000100011100100010100010 0

(String Length =133)

8:

1010111010001000111000111001001100111101010001000100010100010001010
1010010111011001000100010100011001110110001010101000100010001010001
0001010101001010101110111011101110001110101010100100000000000000000
0000000000000000000000000000000000000000000000000000000000101010001
1101110111000100100000000000010101000010010101010001001000000000011
1010000100101011100010010000000000001010100001001010001000000000000
0000010101110111010101110001001000000000000000000000000000000000000
000000000000

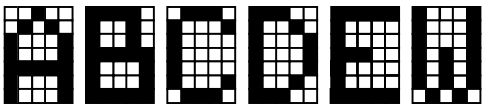
(String length= 481)

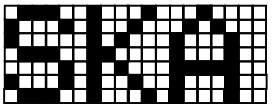
Answers:

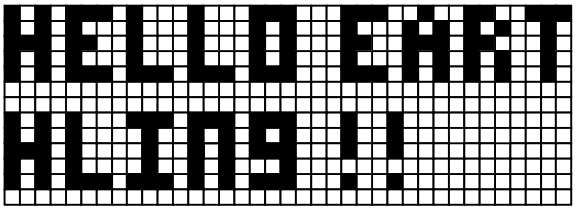
1 – 6. These strings of 35 require 5×7 grids, and yield the letters A B C D E and W.

7. This string requires a 19×7 grid, it yields SKA.

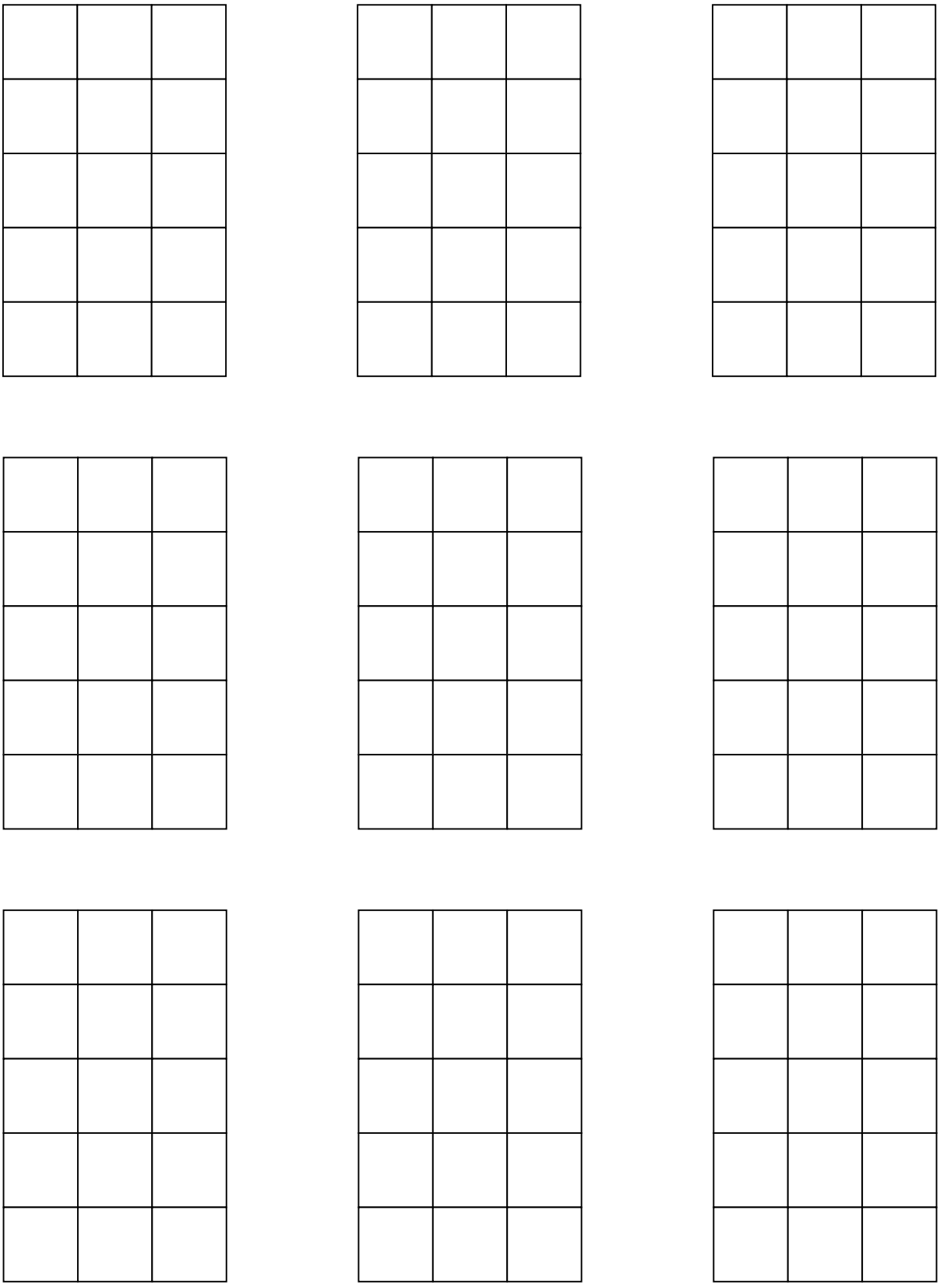
8. This whopper of a string requires a 37×13 grid.







Examples of 15 character strings



C 111101100101111	H 101101111101101	X 101101010101101
9 or g 111101111001111	! 010010010000010	I 111010010010111
F 111100110100100	S 111100111001111	5 111100111001110