Here is a Python program that turns a binary number into an image. The program converts a ‘1’ into a star and a ‘0’ into a space:

```
# get a binary number from the user
img_in = input("Enter your b&w bitmap image: ")
# initially, there is no output
img_out = ""

# loop through each character in the binary input
for character in img_in:
    # add a star (*) to the output if a 1 is found
    if character == "1":
        img_out = img_out + "*
    # otherwise, add a space
    else:
        img_out = img_out + " 

# print the image to the screen
print(img_out)
```

Fill out this table, to record what image is printed when you enter some binary numbers. You can also enter some of your own.

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>11001100</td>
<td></td>
</tr>
<tr>
<td>10101010</td>
<td></td>
</tr>
</tbody>
</table>
Modify your program so that it has a display width of 6 characters. You could create a new variable called “position”, and add 1 to it for every character the user enters, printing a “newline” whenever the position reaches 6.

Your program should now work like this:

```
>>> Enter your b&w bitmap image: 1111110000111111
*****
*
*****
```

Fill out this table, to record what image is printed when you enter some 2-bit binary numbers. How many different shapes can you make?

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>11111110000111111</td>
<td>A hollow rectangle.</td>
</tr>
<tr>
<td></td>
<td>A triangle.</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Can you make a ‘colour’ display, by using 2 binary bits to store each colour?

```
00 = “ “
01 = “.”
10 = “~”
11 = “*”
```

A user could then type in something like “00 01 10 11 01 00”, which would print “.~*~.”

You may need to use this line in your code:

```
for char in img_in.split():
```