

Fill in the functions

Finish the code, run it, and check it.

Name: _____

Date: _____

Part 1 Decode one row

```
def decode_row(clue):  
    row = []          # the row we are building. It starts empty.  
    colour = 0        # we always start with blanks (0)  
  
    # go through each number in the clue (a for loop over the list)  
    for _____ in _____:
```

Write your code here

1. add that many copies of colour (hint: range, row.append)
2. swap the colour: 0 becomes 1, 1 becomes 0

```
    return _____    # what do we give back?
```

```
decode_row([2, 7, 3])  
expected: [0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0]
```

What did YOU get? _____

Did it match? ☐ yes ☐ no, I'll check my code

Part 2 Decode a whole picture

```
def decode_picture(clues):  
    picture = [] # the picture we are building. It starts empty.  
  
    # go through each clue in clues:  
    # 1. decode that clue into a row (use your function!)  
    # 2. add the row to the picture hint: picture.append(...)  
  
    return _____
```

Write your code here

Part 3 Check and time it

Debugging checklist

- ☐ Did I put `return` at the end?
- ☐ Are my indents right?
- ☐ Is colour swapped INSIDE the outer loop but AFTER the inner loop?
- ☐ Did I run the file?

Picture	rows	worked?	time by hand	time by computer
Mystery 1				
Mystery 5				
Mystery 6				
The big one				

Which was faster for the big mystery picture, you or Python? Why?

Stretch: go backwards! Write `encode_row(row)` : count pixels that match the colour, and `append` each count to `clue` . Swap colour when the pixel changes. Don't forget the last count! It should turn `[0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0]` into `[2, 7, 3]` .