Exercises: AES Scheme

We'll only be introducing one new function this week, but it is slightly fiddly on first encountering it. We will be using some more familiar functions too in a slightly more complicated way in order to look at AES encryption up to the MixColumns stage.

Spreadsheet Exercise: AES Partial Encryption

To begin with, download the Excel spreadsheet: `04-AES.xlsx`. The plaintext and key in the spreadsheet are the same as in the lecture, so you can check each stage as we go along.

1. In Cell B1 the plaintext '`tobeornottobethatisthequestion`' is pre-filled. We need the first 16 characters of this in the array B3:E6. The simplest way is to type them in, but we can get the spreadsheet to do this for us automatically. Copy the formula `=MID($B$1,4*(B$2-1)+$A3,1)` into Cell B3 and then copy this across and down to Cell E6. It should display the plaintext characters going down the columns in order. Exercise: Try to see if you can understand what the formula is doing.

$$\begin{bmatrix} t & o & t & e \\ o & r & t & t \\ b & n & o & h \\ e & o & b & a \end{bmatrix}$$

2. In the range B9:E12, use the CODE function and the DEC2HEX functions to turn the plaintext characters in the range B2:E6 into hexadecimal pairs.

3. In the range H3:K6 there is a pre-filled key, and the range H9:K12 below it converts these to text characters for us to use as hexadecimal numbers. The key can be changed in the original range, but nothing needs editing here for now.

4. To save having to `XOR` individual bits of the characters, we will get Excel to do this for each plaintext-key pair. In the range B15:E18 use the HEX2DEC function to change the Array-Hex values to decimal. Do the same for the Key-Hex values but in the range H15:K18.

5. The BITXOR function can now be used to perform the AddRoundKey stage. In range B21:E24 use this function `BITXOR` the decimal plaintext and key values. First, apply the function in Cell B21 to `XOR` the values in

Cell B15 and Cell H15, before using the DEC2HEX function to convert this back to hexadecimal. (‘`=DEC2HEX(BITXOR(plaintext,key),2)`’). Drag/copy this across and down.

6. In order to perform the ByteSub stage, we need to know the row and column for the lookups. In the range B27:E30, use the LEFT function to retrieve the first hex digit from the AddRoundKey values in range B21:E24. Do the same with the RIGHT in the range H27:K30 to retrieve the second hex digit from the AddRoundKey values. We then need to convert these to decimal using HEX2DEC. In both cases, add `+1` to the retrieved values, otherwise the next lookup step won't work correctly. E.g., if the hex pair is 7F, then the row value should return as 8 and the column value should be 16, rather than 7 and 15.

7. We now need to perform the ByteSub stage, which will involve a lookup. We can use the INDEX function, where the array input will be the range `$B$2:$Q$17` in the other worksheet (`Rijndael S-Box`). Use the row and columns values that you just found.

8. Now for the final step, we will again use the INDEX function. The array input should point at the range B33:E36, the row number should be one of the values between A39:A42, and the column number should be from the range H39:K42. This should automatically shift the rows according to those values. You will need to make careful use of absolute referencing to make this work correctly.

9. We won't go any further than this stage, as the MixColumns stage gets quite complicated.