Exercises: RSA

Spreadsheet Exercise: RSA Encryption

To begin with, open your spreadsheet from last week (`05-RSA.xlsx`). Otherwise, download a copy again: `05-RSA.xlsx`.

1. First make sure you are in the worksheet name '`RSA Encryption`'. We will make use of the 'repeated squaring' method in order to handle large powers, working with modular arithmetic.

2. For now, ignore Row 1. In Row 2, there are numbers in Cells B2, D2, and F2. Leave these alone for now.

3. The numbers in the range A4:A19 represent powers and the numbers in the range B4:B19 are the corresponding powers of 2.

4. In Cell C4, set this equal to the value in Cell B2. In Cell C5, use the MOD function to calculate the square of the value in Cell C4, reduced modulo the value in Cell D2. Make sure to use absolute referencing to fix the modulus.

5. We could try to just apply the MOD function to a large power, but things will go wrong. We need to split the power into its binary digits and reduce each squared value as we go along. In Cell F3, enter `=MOD(INT($F$2/B4),2)` to generate the last binary digit of the power. Copy this down to F19. Check that the digits, read upwards, are the binary representation of the denary value in Cell F2. (If you're wondering why we can't just use the DEC2BIN function, try to use this to convert the numbers 511 and 512 to binary. Which one causes a problem?)

6. In the cells G4:G19, we want to return the value in the C column if the value in the F column is 1, but return a 1 if the value in the F column is 0. Use the IF function to do this.

7. Now we can calculate the reduced powers. In Cell H4, use the MOD function to reduce the value in Cell G4 by the modulus in Cell D2. In Cell H5, enter a formula to multiply the value in Cell G5 by the value in Cell H4, reduced by the modulus in Cell D2.

8. At this point you may want to keep a copy of this sheet in this state - this can be done by right-clicking the name of the sheet and clicking `Move or copy...`.

9. The sheet can now be updated to perform RSA calculations. The 'Number' value in Cell B2 should return the `CODE` value of the plaintext character in Cell H1. The primes p and q can be entered as required, along with the encryption exponent e: in the lecture we used primes $p = 11$ and $q = 19$, with $e = 7$. The value in Cell D2 should be the product of p and q, and the power value in Cell F2 should be the value of the encyption exponent e. Update the cells to reflect this.

Spreadsheet Exercise: RSA Decryption

In this part we will use last week's Excel sheet to run the Extended Euclidean algorithm and calculate RSA decryption exponents.

1. If you have last week's spreadsheet premade, then open that. Otherwise, go to last week's instructions for the Euclidean algorithm exercises and follow those.

2. In Cell C1, link this to Cell H5. I.e., use the formula '`=H5`'. Link Cell C2 to Cell H3.

3. Now when you update the values for p, q, and e in Cells H1 - H3, the Euclidean algorithm will run on the encryption exponent e and the totient of the modulus $\varphi(n)$.

4. We need a further step in order to find the decryption exponent d. In Cell E3, use the IF function. If the value in Cell C3 is equal to 1, return the value in B3 with the MOD function applied (using the locked modulus in Cell C1), otherwise return 0 or a blank character.

Practicing RSA

Try the following randomisable question. Make sure that you can perform the calculations by hand and use your spreadsheets to check your answers. **Test**

Yourself Visit the URL below to try a numbas exam:
`https://numbas.mathcentre.ac.uk/question/155037/rsa-encryption/embed/`