Tutorial 3 Solutions

1. *Use the mod 23 encryption system in Lecture 7 to construct a lookup table when the known plaintext is 'P' (14). Use it to find the key when the ciphertext is 'E' (4)*

Multiply the plaintext and key for key values 1 to 22.
1*14=14, 2*14=5, 3*14=19, 4*14=10, 5*14=1, 6*14=15, 7*14=6, 8*14=20, 9*14=11, 10*14=2, 11*14=16, 12*14=7, 13*14=21, 14*14=12, 15*14=3, 16*14=17, 17*14=8, 18*14=22, 19*14=13, 20*14=4, 21*14=18, 22*14=9.
Write as a table with the ciphertext as the primary key, first. Write the ciphertext from 1 to 22 first and then fill in the keys.

1- 5	7-12	13-19	19- 3
2-10	8-17	14- 1	20- 8
3-15	9-22	15- 6	21-13
4-20	10- 4	16-11	22-18
5- 2	11- 9	17-16	
6- 7	12-14	18-21	

Lookup ciphertext 4 to get key 20.

2. *Now construct a time-memory trade off table with 5 chains each of length 5 when the known plaintext is 'P' (14). How many duplicates and how many missing keys are there? Use it to find the key when the ciphertext is 'E' (4)*

Use the work from question 2 to create the chains. We need to pick 5 random keys to start the process.

2 → 5 → 1 → 14 → 12 → 7
4 → 10 → 2 → 5 → 1 → 14
9 → 11 → 16 → 17 → 8 → 20
13 → 21 → 18 → 22 → 9 → 11
15 → 3 → 19 → 13 → 21 → 18

The first 5 numbers in each chain is a potential key. There are 25 in total. 2, 5, 1, 9, 13, 21 are duplicate. 6, 7, 20 are missing.
The table is:

7	2
14	4
20	9
11	13
18	15

Ciphertext 4 is not end value, encrypt it to get 10.
10 is not an end value, encrypt it to get 2.
2 is not an end value, encrypt it to get 5.
5 is not an end value, encrypt it to get 1.
1 is not an end value, encrypt it to get 14.
14 is an end value and so the key is in the chain starting with 4.

We encrypted the ciphertext 5 times to find the value in the table. We must encrypt the starting value 'chain length' – 5 = 5 -5 = 0 times to get the key. The key is 4.

3. *How is encryption and decryption performed with a Feistal cipher? Prove that decryption undoes encryption.*

A Feistal cipher starts with an initial permutation, then performs a number of encryption steps with a sub-key which is generated from the encryption key. It then performs the reverse permutation. Each step splits the data into left and right halves. The new left half is the old left half xored with a function of the right half and the sub-key, while the right half remains unchanged. The two halves are then swapped. Decryption performs the same steps but with the sub-keys in the reverse order.

The initial permutation undoes the reverse of the initial permutation. Swapping twice does not change the data. Xor with the same function twice has no effect. The keys are used in the reverse order so that the last encrypted step is undone by the first decrypted step because they are both using the same key. This applies to all the steps.

4. *Describe briefly how the DES algorithm implements the Feistel scheme, mentioning in particular the role of the key length and also the so called S-Boxes. You do not need to provide details of the actual S-Boxes, rather describe in general terms their role in the algorithm.*

DES has 16 complex steps, alternating with simple permutations. Separate sub keys are used for each of the complex steps. It has a key size of 56 bits, reduced from the original 128 bits after advice from the USA NSA. The S-Boxes are just look up tables, designed for good confusion and diffusion.