Fall 2022 CIS 3362 Homework #4 Solutions

1) (20 pts) Consider a simple cryptosystem with a 16 bit block and 16 bit key as follows:

Step 1: Compute $C_0 = A(P)$, where A is the permutation matrix shown below.
Step 2: Compute $C_1 = C_0 \oplus K$, where K is the input key.
Step 3: Compute $C_2 = S(C_1)$. S splits its input into four blocks of four bits. For each block of four bits, it substitutes for it values shown in B. For example, if the four bits of input were 0101, which corresponds to 5, we would look in spot #5 in B, in row 1, column 2, which is 12 and substitute 1100. As another example if the input were 1110, the output would be 0101.
Step 4: Let $C_2 = LR$, where L is the left byte and R is the right byte. Compute the ciphertext C = RL.

$$A = \begin{bmatrix} 3 & 7 & 12 & 9 \\ 11 & 14 & 6 & 1 \\ 15 & 16 & 10 & 13 \\ 2 & 4 & 5 & 8 \end{bmatrix} \qquad B = \begin{bmatrix} 6 & 1 & 11 & 4 \\ 13 & 12 & 15 & 8 \\ 0 & 3 & 10 & 9 \\ 7 & 2 & 5 & 14 \end{bmatrix}$$

Compute the encryption of the plaintext A349 with the key 5DF7.

```
 P: 1010 0011 0100 1001

C0: 1100 0001 0111 0001
 K: 0101 1101 1111 0111
C1: 1001 1100 1000 0110

C2: 0011 0111 0000 1111

 C: 0000 1111 0011 0111
```

2) (4 pts) If the input in DES to S-box 5 is 001111, what is the output?

row = 01 = 1
col = 0111 = 7
$S_5(1,7) = 1$
0001

3) (8 pts) The first part of the function F in a round of DES expands the 32-bit input (from the right half of the previous round) to 48 bits. If this input, in HEX to the function F is 7DA839B2, what are the last 8 bits of output right after this value is processed by the Expansion Permutation E?

These last 8 bits are numbered: 28, 29, 28, 29, 30, 31, 32, 1

The first bit is 0 since 7 = 0111.
Bits 29 – 32 are 0010.
Bit 28 is 1 since B is 1011.

Taking the appropriate bits, we get 10100100.

4) (10 pts) Without examining all entries in the 16 round key schedule of DES, determine whether or not each number (which represents a bit location in the original key in each of the 16 boxes labeled "Round 1" through "Round 16") appears the exact same number of times collectively in the 16 boxes. (As an example, 10 appears in round except rounds 4, 12 and 14, so it appears 13 times.) Give proof of your answer.

Each number does not appear the same number of times in the 16 boxes. To see this, note that there are 48 values in each box, so there are a total of 16 x 48 values. Each value is one of 56 possible ones, thus the average number of times each value appears is $\frac{16\times48}{56} = \frac{16\times6}{7} = \frac{96}{7}$. Since this number isn't an integer, it's impossible that each of the values appears exactly this many times. Instead, it must be the case that some values appear more than this number of times while other values appear less than this. (This number is in between 13 and 14. So 10 appears slightly less often than the average number.)

5) (15 pts) One mathematical idea that has come up multiple times in this course is matrix multiplication. To multiply two square matrices of size n x n, our answer will also be a matrix of size n x n. To compute the answer in row i, column j, we take each item in row i of the first matrix, multiply it by the corresponding item in column j of the second matrix and add these n products to get the desired result. Complete the C function below to multiply the two input matrices mat1 and mat2 of size 10 x 10 and store the result in mat3. You may assume that mat1 and mat2 store the desired values before the function is called. Do not worry about integer overflow.

```
void matMult(int mat1[][10], int mat2[10], int mat3[][10]) {

    int i, j, k;

    for (i=0; i<10; i++) {

        for (j=0; j<10; j++) {

            mat3[i][j] = 0;
            for (k=0; k<10; k++)
                mat3[i][j] += (mat1[i][k]*mat2[k][j]);
        }
    }
}
```

6) (8 pts) In the key expansion algorithm of AES, if w[30] = 79AEC508 and w[27] = FD1B6423, what is w[31]?

```
w[30] = 0111 1001 1010 1110 1100 0101 0000 1000
w[27] = 1111 1101 0001 1011 0110 0100 0010 0011
xor   = 1000 0100 1011 0101 1010 0001 0010 1011
HEX   = 8    4    B    5    A    1    2    B
```

7) (20 pts) Let the input to the MixCols (during AES encryption) be $\begin{bmatrix} A0 & 74 & 65 & B7 \\ 2B & 8D & 2E & C6 \\ 99 & 1F & C8 & EB \\ C5 & E5 & F7 & 23 \end{bmatrix}$.

What's the output in row 2 col 4? (The matrix by which to "multiply" is $\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}$.)

We must compute

01 x B7 = 1011 0111 (no change to original)
02 x C6 = 1001 0111 (work below)
03 x EB = 0010 0110 (work below)
01 x 23 = 0010 0011 (no change to original)

Ans = 0010 0101 (25)

C6 = 1100 0110
2 x C6 = 1 1000 1100

Mapped = 1000 1100
xor 1 1011
2 x C6 = 1001 0111

EB = 1110 1011
2 x EB = 1 1101 0110
Mapped = 1101 0110
xor 1 1011
2 x EB = 1100 1101

EB = 1110 1011
2 x EB = 1100 1101
3 x EB = 0010 0110

8) (15 pts) Consider an AES plaintext of $\begin{bmatrix} 01 & 89 & FE & 76 \\ 23 & AB & DC & 54 \\ 45 & CD & BA & 32 \\ 67 & EF & 98 & 10 \end{bmatrix}$ with a key of 128 1s. Show the state matrix after the shift rows step in Round 1.

First, we add the key, which means inverting every byte, to obtain: $\begin{bmatrix} FE & 76 & 01 & 89 \\ DC & 54 & 23 & AB \\ BA & 32 & 45 & CD \\ 98 & 10 & 67 & EF \end{bmatrix}$.

Next we subbytes to get: $\begin{bmatrix} BB & 38 & 7C & A7 \\ 86 & 20 & 26 & 62 \\ F4 & 23 & 6E & BD \\ 46 & CA & 85 & DF \end{bmatrix}$.

Finally, shift rows to get: $\begin{bmatrix} BB & 38 & 7C & A7 \\ 20 & 26 & 62 & 86 \\ 6E & BD & F4 & 23 \\ DF & 46 & CA & 85 \end{bmatrix}$.