Mission 6 **Name:**_____

Use the Sage code demonstrated in class to attack an SDES system. Do all of your work in the Mission 6 CoCalc Assignment files. (They will be collected as well.)

Part 1: Use differential cryptanalysis to attack SDES (3 rounds).

You encrypt the plaintext P=[0,0,0,0,1,1,1,1,1,0,1,1] and get the ciphertext C=[0,1,0,0,1,1,1,1,1,1,0,0]. In particular L_3 = [0,1,0,0,1,1] and R_3=[1,1,1,1,0,0].

In order to attack the system, you also use several values for a second plaintext, P*. The values of the plaintext, along with the ciphertext and the corresponding values of the xor of the inputs to the sboxes, $(E(L_3) \oplus E(L^*_3))$ and the outputs $((R_3 \oplus R_3^*) \oplus (L_0 \oplus L_0^*))$ are given. Determine the value of K_3.

First alternate plaintext: P* = [1,0,0,1,1,0,1,1,1,0,1,1]

$(E(L_3) \oplus E(L^*_3))$ = [1,1,1,0,1,0,0,1] $((R_3 \oplus R^*_3) \oplus (L_0 \oplus L_0^*))$ = [1,0,0,1,0,1]

Possible values of input to Sbox 1 (From L_0):

Possible values of input to Sbox 2 (From L_0):

Second alternate plaintext: P* = [0,1,0,1,1,1,1,1,1,0,1,1]

$(E(L_3) \oplus E(L^*_3))$ = [1,0,0,0,0,0,1,1] $((R_3 \oplus R^*_3) \oplus (L_0 \oplus L_0^*))$ = [0,1,1,0,1,1]

Possible values of input to Sbox 1 (From L_0):

Possible values of input to Sbox 2 (From L_0):

Third alternate plaintext: P* = [0,1,1,0,1,1,1,1,1,0,1,1]

$(E(L_3) \oplus E(L^*_3))$ = [1,1,0,1,0,1,0,1] $((R_3 \oplus R^*_3) \oplus (L_0 \oplus L_0^*))$ = [1,1,0,1,1,1]

Possible values of input to Sbox 1 (From L_0):

Possible values of input to Sbox 2 (From L_0):

Conclude that the input to the Sboxes from the original plaintext was:

Input=_____(Concatenate the remaining values for the input to Sbox 1 and 2.)

We can now recover the value of K_3 by xoring this string with the value of $E(L_3)$:

$E(L_3)$:_____ $\oplus$ Input:_____ =K3 :_____

Part 2: Use a meet in the middle attack to recover the two keys K_1 and K_2 used in an implementation of 2SDES (Double encryption with SDES) using 4 rounds.

First you encrypt P=[0,1,0,1,0,1,0,1,0,1,0,1] and get C=[0, 1, 0, 1, 1, 1, 1, 0, 0, 1, 1, 0].

a. Use a brute force attack to find all possible values of K_1 and K_2. How many seconds does it take?

b. Use the Meet-In-The-Middle attack to find the same information. Are they the same as the ones you found by brute force?

c. How many seconds did this take?

d. Explain briefly why this was so much faster, how many encryptions does each method require? How many times faster would you expect a meet in the middle attack to be in this situation? (Recall, the keys have 9 bits...)

Now you find that encrypting P*=[0, 0, 1, 0, 0, 1, 0, 0,1, 0, 0, 1] with the same keys produces the ciphertext C*=[1, 1, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0].

e. Repeat the meet in the middle attack, and compare the pairs of keys you got using P and C to obtain the binary values of K_1 and K_2:

f. Use the int2bin function to convert these numbers back into binary and record them here:

K_1=_____

K_2=_____