Problem 1.

There is a dice. In each event, you throw the dice and observe the number.

a. You are using a typical dice. What is the information entropy value for one event? What

is the entropy value for four events?

**H = log(sn) = nlog(s)< this is in base 2. Dice have equal probability of all numbers.**

**One event = 2.58**

**Four events = 10.32**

b. Now suppose you modify the dice so that the side that originally showed number six

becomes a five (the numbers 1, 2, 3, 4 occupy one side each, and 5 occupies two sides).

What is the information entropy for one event? What is the entropy for four events?

4(1/6log(6))+1/3(log(3)) = 2.44

**One event = 2.44**

**Four events = 9.76**

c. You can continue modifying the dice. How would you maximize the entropy? Also, how

would you minimize the entropy?

**To maximize you would have an n sided die with no repeating numbers. To minimize the entropy you would have the dice have all the same number.**

Problem 2.

a. Determine gcd(24140,16762).

**=GCD(16762,MOD(24140,16762)) = 34 what I used in excel**

b. Determine gcd(4655,12075).

**=GCD(12075,MOD(4655,12075)) = 35 what I used in excel**

c. Determine gcd(4278,8602).

**=GCD(8602,MOD(4278,8602)) = 46**

Problem 3. Prove the following:

1. a ≡ b (mod n) implies b ≡ a (mod n)

If a, b ∈ Z and a ≡ b (mod n), then a − b = k · n for some k ∈ Z, and so b − a = (−k) · n, so that b ≡ a (mod n) also. Hence proved

1. a ≡ b (mod n) and b ≡ c (mod n) imply a ≡ c (mod n)

If a, b, c ∈ Z, with a ≡ b (mod n) and b ≡ c (mod n), then a − b = k · n and b − c = l · n, for some k, l ∈ Z, so that a − c = a − b + b − c = (k + l) · n, and a ≡ c (mod n). Hence proved

1. [(a mod n) - (b mod n)] mod n = (a - b) mod n

let a=hn-(a mod n),b=kn-(b mod n) and h,k belong to Z  
then the R.H.S  
(a-b) mod n=[ a-b-(h+k)n] mod n  
=[hn- amod n )+(kn+b mod n)-hn -kn]mod n  
=(a mod n- b mod n) mod n=L.H.S  
Hence proved

d. [(a mod n) × (b mod n)] mod n = (a × b) mod n

Hence proved.

Problem 4.

Prove the following:

1. For two consecutive integers n and n+1, gcd(n,n+1)=1

So d/n and d/n+1 so d/n+1-n therefore d/1 and d is equal to +or - 1

1. Given two integers a and b, prove that Euclidean algorithm, described in Section 2.2, yields the greatest common divisor gcd(a,b)

Suppose a and b are integers with a ≥ b > 0.

(1) Apply the division algorithm: a = b q + r, 0 ≤ r < b.

(2) Rename b as a and r as b and repeat until r = 0.

The last nonzero remainder is the greatest common divisor of a and b.

The Euclidean Algorithm depends upon the following lemma.

If a = bq + r, then GCD(a, b) = GCD(b, r).

Proof. We will show that if a = bq + r, then an integer d is a common divisor of a and b if, and only if, d is a common divisor of b and r.

Let d be a common divisor of a and b.

Then d|a and d|b.

Thus d|(a − bq), which means d|r, since r = a − bq.

Therefore d is a common divisor of b and r.

Now suppose d is a common divisor of b and r. Then d|b and d|r.

Therefore d|(bq +r), so d|a.

Therefore, d must be a common divisor of a and b.

Therefore , the set of common divisors of a and b are the same as the set of common divisors of b and r.

It follows that d is the greatest common divisor of a and b if and only if d is the greatest common divisor of b and r

**Problem 5**

The idea of Feistel networks is that the decryption algorithm is the same as the encryption one, only using the subkeys in the reverse order.

However, the subkey generation algorithm described in the problem the reverse order of subkeys is the same as the original one.

Thus the encryption algorithm used as a black box decrypts ciphertext.

On a more detailed level a Feistel network divides an input string into two halves, L0, R0, and then updates the halves using the following rules:

On round i it starts with strings Li−1, Ri−1 and sets Li = Ri−1, Ri = Li−1 ⊕ F(Ri−1, Ki), where F is a round function (fixed, realized through S-boxes), and Ki is the i-th round subkey.

Now, we have to show that Ri−1 = Li , Li−1 = Ri ⊕ F(Li , Ki).

(This means that applying the algorithm and swapping the two halves we in the end obtain the plaintext.)

The first equation is obvious; to obtain the second we observe that Li = Ri−1, and so F(Li , Ki) = F(Ri−1, Ki ).

**Problem 6**

a) in binary notation: 0000 1011 0000 0010 0110 0111

1001 1011 0100 1001 1010 0101

in hexadecimal notation: 0 B 0 2 6 7 9 B 4 9 A 5

b) L0, R0 are derived by passing the 64-plaintext through Initial Permutation:

L0 = 1100 1100 0000 0000 1100 1100 1111 1111

R0 = 1111 0000 1010 1010 1111 0000 1010 1010

c) EXP(R0) = 011110 100001 010101 010101 011110 100001 010101 010101

d) A = 011100 010001 011100 110010 111000 010101 110011 110000

e) 0 (base 10)=0000 (base 2), 12 (base 10)=1100 (base 2), 2 (base 10)=0010 (base 2), 1 (base 10)=0001 (base 2), 6 (base 10)=0110 (base 2), 13 (base 10)=1101 (base 2), 5 (base 10)=0101 (base 2), 0 (base 10)=0000 (base 2)

f)B = 0000 1100 0010 0001 0110 1101 0101 0000

g) P(B) = 1001 0010 0001 1100 0010 0000 1001 1100

h) R1 = 0101 1110 0001 1100 1110 1100 0110 0011

i) L1 = R0. The ciphertext is the concatenation of L1 and R1.