**Assignment Solutions 3**

1)

n = 143; φ(n) = 120; d = 11; C = 106.

2)

n = 527; φ(n) = 480; d = 343; C = 128. For decryption, we have 128343 mod 527 = 128256 × 12864 × 12816 × 1284 × 1282 × 1281 mod 527 = 35 × 256 × 35 × 101 × 47 × 128 = 2 mod 527 = 2 mod 257.

3) By trail and error, we determine that p = 59 and q = 61. Hence φ(n) = 58 x 60 = 3480. Then, using the extended Euclidean algorithm, we find that the multiplicative inverse of 31 modulo φ(n) is 3031.

4) No, it is not safe. Once Bob leaks his private key, Alice can use this to factor his modulus, N. Then Alice can crack any message that Bob sends.

Here is one way to factor the modulus:

Let k= ed – 1. Then k is congruent to 0 mod f(N) (where 'f is the Euler totient function). Select a random x in the multiplicative group Z(N). Then xk ≡ 1 mod N, which implies that xk/2 is a square root of 1 mod N. With 50% probability, this is a nontrivial square root of N, so that gcd(xk/2 – 1,N) will yield a prime factor of N.

If xk/2 = 1 mod N, then try xk/4, xk/8, etc...

This will fail if and only if xk/2^i ≡ –1 for some i. If it fails, then choose a new x.

This will factor N in expected polynomial time.

5)

------------------------------------------

| Square-and-multiply algorithm (L-to-R) |

------------------------------------------

n = 221;

x = m = 2, e = 11 = (1011)\_2;

i=2. e[2]=0, x = x^2 = 4;

i=1. e[1]=1, x = x^2 = 16, x = x\*m = 32;

i=0. e[0]=1, x = x^2 = 140, x = x\*m = 59;

The final result is:

m^e mod n = 2^11 mod 221 = 59.

------------------------------------------

| Square-and-multiply algorithm (R-to-L) |

------------------------------------------

n = 221;

x = 1, y = 2, e = 11 = (1011)\_2;

i=0. e[0]=1, x = x\*y = 2, y = y^2 = 4;

i=1. e[1]=1, x = x\*y = 8, y = y^2 = 16;

i=2. e[2]=0, y = y^2 = 35;

i=3. e[3]=1, x = x\*y = 59, y = y^2 = 120;

The final result is:

m^e mod n = 2^11 mod 221 = 59.

------------------------------

| Montgomery Powering Ladder |

------------------------------

x = 1, y = 2

e = 11 = (1011)\_2;

Init: x = 1; y = 0;

i=3. e[3]=1: x = 2; y = 0;

i=2. e[2]=0: x = 4; y = 0;

i=1. e[1]=1: x = 32; y = 0;

i=0. e[0]=1: x = 59; y = 0;

The final result is:

m^e mod n = 2^11 mod 221 = 59.

6) The statement is false. Such a function cannot be one-to-one because the number of inputs to the function is of arbitrary, but the number of unique outputs is 2^n. Thus, there are multiple inputs that map into the same output.

7) There are two properties for a good cryptographic hash function, namely one-way function and collision free.

In the following we will check whether or not the given hash function satisfies the two properties.

* For the property of being a one-way function:
  + It is easy to compute h(x) for given x: first compute   
    (-x) mod 2159 and write this intermediate result in binary form, then change 0 to 1 at its bit position 2159.
  + However, the inverse function can also be solved fairly easily for some cases. For example, if h(x)= 2159 + 8, then the input x can be solved as x = 2159 – 8.
  + So it does not meet the requirement of the first property.
* For the property of collision free:
  + It is not difficulty to see that both x1= 2159–8 and x2= 2160–8 give the same hashing result as h(x1) = h(x2) = 2159 + 8.
  + So it does not meet the requirement of the second property either.
* In summary, this is not a good cryptographic hash function.

8) The CBC mode with an IV of 0 and plaintext blocks D1, D2, . . ., Dn and 64-bit CFB mode with IV = D1 and plaintext blocks D2, D3, . . ., Dn yield the same result.