1. **This problem is about ElGamal encryption and signature schemes. (30 points)**

(a) Let $p = 83$ and $g = 16$ be a generator of Z_{83}^*. Assume that the public key is $(p, g, 59)$ and the secret key $(p, g, 29)$. Encrypt the plaintext $m = 25$ and decrypt the ciphertext $(56, 13)$.

Ans:

$p = 83,\ g = 16\ and\ y = 59$

To encrypt the plaintext $m = 25$, we need to choose a random integer k, compute the ciphertext pair (c_1, c_2), and send it to the recipient.

Let's follow the encryption steps:

Choose a random integer k. Let's say $k = 2$.

Compute the first part of the ciphertext c_1.

$$c_1 = g^k \bmod p = 16^2 \bmod 83 = 7$$

Compute the second part of the ciphertext c_2

$$c_2 = g^k \bmod p = 25 * 59^2 \bmod 83 = 41$$

the ciphertext pair $(c_1, c_2) = (7, 41)$

To decrypt the ciphertext (56, 13) using the secret key (p, g, x) = (83, 16, 29), let's follow the decryption steps:

Compute the shared secret key.

$$s = {c_1}^k \bmod p = 56^{29} \bmod 83 = 6$$

Compute the multiplicative inverse of s.

$$s^{-1} = 14\ since\ 6 * 14 \bmod 83 = 1$$

Decrypt the ciphertext c2.

$$m = (13 * 14) \bmod 83 = 16$$

The decrypted plaintext is $m = 16$.

(b) Use the secret key as the signing key to sign the message $m = 25$. The randomly chosen k *is* 23. You don't need to do hashing before signing.

Ans:

$$Given: p = 83, g = 16, x = 29, m = 25, k = 23$$

Compute the first part of the signature r.

$$r = g^k \bmod p = 16^{23} \bmod 83 = 28$$

Compute the second part of the signature s.

$$s = k^{-1} * (m - x * r) \bmod (p - 1)$$
$$= 23^{-1} * (25 - 29 * 28) \bmod (83 - 1)$$
$$= 25 * (-787) \bmod 82$$
$$= (-19675) \bmod 82 = 5$$

The signature pair is (r, s) = (28, 5).

2. **For DSA, let the public key be $(p = 149, q = 37, g = 41, y = 144)$, and the secret key be $(p = 149, q = 37, g = 41, x = 26)$. Assume that the hash function is $h(m) = m^{21}\ mod\ 37$. (30 points)**

(a) Compute the signature of $m = 9876543210$.

Ans:

Given $pk = (149, 37, 41, 144)\ and\ sk = (149, 37, 41, 26)$

Let's choose $k = 2$

$$r = \left(g^k\ mod\ p\right)\ mod\ q$$
$$= (41^2\ mod\ 149)\ mod\ 37$$
$$= 42\ mod\ 37 = 5$$

Compute the hash of the message.

$$h(m) = 9876543210^{21}\ mod\ 37 = 1$$

Compute the second part of the signature, s.

$$s = (h(m) + x * r) * k^{-1}\ mod\ q$$
$$= (1 + 26 * 5) * 2^{-1}\ mod\ 37$$
$$= 131 * 2^{-1}\ mod\ 37$$
$$= 131 * 19\ mod\ 37 = 10$$

The signature pair is $(r, s) = (5, 10)$.

(b) Is $(12, 25)$ a valid signature for $m = 3248$?

Ans:

Given $pk = (149, 37, 41, 144)$, signature pair is $(r, s) = (12, 25)$

Compute the hash of the message.

$$h(m) = 3248^{21}\ mod\ 37 = 31$$

Compute w, the modular multiplicative inverse of s mod q.

$$w = s^{-1}\ mod\ q$$
$$= 25^{-1}\ mod\ 37 = 3$$

Compute v:

$$v = \left(g^{h(m)}y^r\right)^w\ mod\ p\ mod\ q$$
$$= ((41^{31} * 144^{12})^3\ mod\ 149)\ mod\ 37$$
$$= 65\ mod\ 37 = 28$$

In this case, $v = 28\ and\ r = 12$.

Since $v \neq r$, the signature $(12, 25)$ is not valid for the message $m = 3248$.

Therefore, the given signature is not valid for the provided message.

3. **Why is the "sequential" DL interactive proof system zero-knowledge? Why isn't**
   the "parallel" FS interactive proof system zero-knowledge? (20 points)

   Ans:

   Because the prover cannot acquire any information without interacting with the verifier, the sequential system is considered to be a zero-knowledge system. The parallel system is not a zero-knowledge system since the prover is able to acquire information by just monitoring the output of the verifier.

4. **We consider the multi-authority secure electronic voting scheme without a trusted center, discussed in classes. How does the authority A_i assures A_j that the sent share $s_{i,j} = f_i(x_j)$ is indeed consistent with all other shares sent to the other authorities? (20 points)**

   Ans:

   Security rests on three principles: each authority must be able to check consistency of its share as it is sent and later compared against other shares; each authority must receive a secret key from which it can verify its own shares if required; any authority which fails to follow these rules must lose its right to participate in future votes - this incentivizes each member of the group to play fair. We examine such schemes in our setting of n authorities and m voters, where each voter has a secret i,i,d. sequence x_i $for\ i = 1, 2, \cdots, m$. The authority sends two messages: share $f_i(x_j)$ and challenge c(r), where r=1, f1=2m. He knows $r(f_i)$ should be correct in this case.