1. Can you identify the (possible) weaknesses of this digital signature scheme? Can you compose a different message m' such that $h(m) = h(m')$?

   Solution: The weakness of our scheme lies in the hash function, which computes the same hash values for different messages, and in fact it is easy to find messages with the same hash value—for example, by adding pairs of letters (anywhere in the message) such that their corresponding ASCII values are inverses modulo p.

   Examples (from the assignments) of messages with the same hash value are: "A mess" and "L message." In general, by its nature, any hash function is going to have such *collisions*, i.e., messages such that:

   $$h(\text{A message.}) = h(\text{A mess}) = h(\text{L message}) = 5,$$

   but there are hash functions which are *collision-resistant* in the sense that it is computationally hard to find two messages m, m' such that $h(m) = h(m')$. A good hash function is also a *one-way function* in the sense that given a value y it is computationally hard to find an m such that $h(m) = y$.

2. If you receive a message m, and a signature pair (r, s), and you only know p, g and $y = g^x \pmod p$, i.e., p, g, y are the *public* information, how can you "verify" the signature—and what does it mean to verify the signature?

   Solution: Verifying the signature means checking that it was the person in possession of x that signed the document m. Two subtle things: first we say "in possession of x" rather than the "legitimate owner of x," simply because x may have been compromised (for example stolen). Second, and this is why this scheme is so brilliant, we can check that "someone in possession of x" signed the message, even *without knowing what x is*! We know y, where $y = g^x \pmod p$, but for large p, it is difficult to compute x from y (this is called the Discrete Log Problem, DLP).

   Here is how we verify that "someone in possession of x" signed the message m. We check $0 < r < p$ and $0 < s < p - 1$ (see Q6), and we compute $v := g^{h(m)} \pmod p$ and $w := y^r r^s \pmod p$; g, p are public, m is known, and the function $h : \mathbb{N} \longrightarrow [p-1]$ is also known, and r, s is the given signature. If v and w match, then the signature is valid.

   To see that this works note that we defined $s := k^{-1}(h(m) - xr) \pmod{p-1}$. Thus, $h(m) = xr + sk \pmod{p-1}$. Now, Fermat's Little Theorem (FLT—see page 114 in the textbook), says that $g^{p-1} = 1 \pmod p$, and therefore

   $$g^{h(m)} \overset{(*)}{=} g^{xr+sh} = (g^x)^r (g^k)^s = y^r r^s \pmod p.$$

   The FLT is applied in the $(*)$ equality: since $h(m) = xr + sk \pmod{p-1}$ it follows that $(p-1) | (h(m) - (xr + sk))$, which means that $(p-1)z = h(m) - (xr + sk)$ for some z, and since $g^{(p-1)z} = (g^{(p-1)})^z = 1^z = 1 \pmod p$, it follows that $g^{h(m)-(xr+sk)} = 1 \pmod p$, and so $g^{h(m)} = g^{xr+sk} \pmod p$.

3. Research on the web a better suggestion for a hash function h.

   Solution: Here are the hash functions implemented by `GPG`, version 1.4.9: `MD5, SHA1, RIPEMD160, SHA256, SHA384, SHA512, SHA224.`

4. **[Bonus Question]** Show that when used without a (*good*) hash function, ElGamal's signature scheme is *existentially forgeable*; i.e., an adversary Eve can construct a message m and a valid signature (r, s) for m.

   Solution: To see this, let b, c be numbers such that $\gcd(c, p-1) = 1$. Set $r = g^b y^c$, $s = -rc^{-1} \pmod{p-1}$ and $m = -rbc^{-1} \pmod{p-1}$. Then (m, r, s) satisfies $g^m = y^r r^s$. Since in practice a hash function h is applied to the message, and it is the hash value that is really signed, to forge a signature for a meaningful message is not so easy. An adversary has to find a meaningful message $\tilde{m}$ such that $h(\tilde{m}) = h(m)$, and when h is collision-resistant this is hard.

5. **[Bonus Question]** In practice k is a random number; show that it is absolutely necessary to choose a new random number for each message.

   Solution: If the same random number k is used in two different messages $m \neq m'$, then it is possible to compute k as follows: $s - s' = (m - m')k^{-1} \pmod{p-1}$, and hence $k = (s - s')^{-1}(m - m') \pmod{p-1}$.

6. **[Bonus Question]** Show that in the verification of the signature it is essential to check whether $1 \leq r \leq p-1$, because otherwise Eve would be able to sign a message of her choice, provided she knows one valid signature (r, s) for some message m, where m is such that $1 \leq m \leq p-1$ and $\gcd(m, p-1) = 1$.

   Solution: Let m' be a message of Eve's choice, $u = m'm^{-1} \pmod{p-1}$, $s' = su \pmod{p-1}$, r' and integer such that $r' = r \pmod{p}$ and $r' = ru \pmod{p-1}$. This r' can be obtained by the so called Chinese Reminder Theorem (see 117 in the textbook). Then (m', r', s') is accepted by the verification procedure.