1. **(10 points) Explain the relationship between Pre-image resistance and Strong Collision resistance. Give examples to support your answers.**

Pre-image resistance means that given h(x) it is hard to find x, it is one way.

Strong collision resistance means that given any information it is hard to find x.

Preimage resistance is part of strong collision resistance. As a strong collision resistance implies one way.  
An example for preimage resistance is a phone book, all names are in alphabetical order. Given a name it is easy to find a phone number but given a phone number it is hard to find a name associated with it.

On example of a strong collision resistance is a hash database where every entrée has a unique hash code, ie no collisions.

1. **(10 points) When encrypting and signing a message m, does the order of encryption and signature operations matter? Explain.**

The order of operations matters when different tasks are needed to be accomplished. If the sender encrypts then signs the message then anyone with access to the encrypted message can know who sent it and confirm the authenticity. On the other hand if the sender signed then encrypted the message the only one who can confirm the authenticity is the intended receiver with the decryption key.

1. **(10 points) Which one of the following Hash function requirements is used to prevent forgery when an encrypted hash code is used? Justify your answer**

**b.** Second preimage resistance

by definition it makes it computationally impossible to find other messages with the same hash value as the message being sent.

1. **(20 points)**
   1. This function satisfies the first three properties but falls flat on the 4th one. It would be feasible for different messages to have the same hash value.
   2. As in the above it would steel by possible to find a similar hash function with different messages.
   3. (189+632+900+722+349) ^ 2 % 989 = 955
2. **(10 points) State the value of the padding field in SHA-512 if the length of the message is** 
   1. **1919 bits**

(896-1919) % 1024 = 1 bit

* 1. **1921 bits.**

(896-1921) % 1024 = 1023 bits

1. **(20 points)**
   1. **i.** Yes, it will be detected by Bob. When the message gets sent by Alice there is only one hash value generated which then would be altered by Oscar. When Bob receives the message he would see that the hash values are in fact different, detecting the interference.

**ii.** Yes, it will be detected by Bob. The checksum generated will be different from the one received.

* 1. **i.** No, since Oscar knows the signature it will not be detected.

**ii.** No, since Oscar knows the signature it will not be detected.

* 1. **i.** Yes, since both claim that they send the message, all Bob has to do is compare the signature he has to the one provided by both Alice and Oscar. In this case Oscars signature will be different from the one in the message.

**ii.** Yes, here Bob could do the same as above while utilizing a checksum.

* 1. **i.** If Bob would forward the message to Alice she can compare it to her signature, if it is different then she can prove that she did not send the message.

**ii.** Here it would not be possible since both parties have the signature.