Problem 1: Malleability of ElGamal

Remember the auction example from the lecture: Bidder 1 produces a ciphertext \( c = E(pk, bid_1) \) where \( E \) is the ElGamal encryption algorithm (using integers mod \( p \) as the underlying group). Given \( c \), Bidder 2 can then compute \( c' \) such that \( c' \) decrypts to \( 2 \cdot bid_1 \mod p \). This allows Bidder 2 to consistently bid twice as much as Bidder 1.\(^1\)

Now refine the attack. You may assume that \( bid_1 \) is the amount of Cents Bidder 1 is willing to pay. And you can assume that Bidder 1 will always bid a whole number of Euros. (I.e., \( bid_1 \) is a multiple of 100.)

Show how Bidder 2 can consistently overbid Bidder 1 by only 1%. What happens to your attack if Bidder 1 suddenly does not bid a whole number of Euros?

**Hint:** Remember that modulo \( p \), one can efficiently find inverses. For example, one can find a number \( a \) such that \( a \cdot 100 \equiv 1 \mod p \).

Problem 2: Hybrid encryption – implementations

Implement a hybrid encryption using ElGamal and AES. You are allowed to use ready-made ElGamal and AES.

In the contributed file `hybrid.py` (lecture webpage), you find a prepared template in Python that already provides function for ElGamal and AES encryption as well as some utility functions and testing code that you might need. I recommend to use that code. If you wish to use another language, you will have to find your own ElGamal and AES routines.

You should check that `hybrid_decrypt(sk, hybrid_encrypt(pk, msg))` returns `msg`.

It is OK if you only allow encrypting messages whose length is a multiple of 16 bytes (blocklength of AES).

Problem 3: Hash functions

Let \( E \) be a block cipher with key and block length \( n \). Let \( F(x\|y) := E(x, y) \) (a hash function with fixed input length \( 2n \), called a compression function).

Show how to find a collision for \( F \). (I.e., break collision-resistance of \( F \).)

---

\(^1\)As long as \( bid_1 < p/2 \), that is. Otherwise \( 2 \cdot bid_1 \mod p \) will not be twice as much as \( bid_1 \). However, for large \( p \), \( bid_1 \geq p/2 \) is an unrealistically high bid.