Exercise 1 (Hash Collisions)

1. Find a CRC-32 collision with the string “Satoshi Nakamoto”.
2. Find a SHA-256 collision with the string “Satoshi Nakamoto”.

code/attack.py

```python
# for sha256 use:
# from hashlib import sha256
# sha256(x).digest() instead of crc32(x)

from binascii import crc32
from sys import byteorder

x = b"Satoshi Nakamoto"
hx = crc32(x)

for n in range(2 ** 64):

    y = n.to_bytes(8, byteorder)

    if n % 10 ** 7 == 0:
        print(f"round {n:,} y: {y.hex()}")

    if crc32(y) == hx:
        print(f"Collision found after {n:,} rounds: {x},{y}")
        break


# Collision found after 3,311,842,240 rounds:
# b'Satoshi Nakamoto','c0b366c500000000'
```

Exercise 2 (Signature Scheme)

1. Install a signature scheme. A signature scheme I recommend is PyNaCl.
2. Generate a keypair.
3. Sign a message.
4. Verify the signed message.
5. Tamper with the signed message and check that verification fails.

Exercise 3 (Sign using Bitcoin)

Use the bitcoin client to sign a message, send the signed message to your partner by email, and let him verify the message.