Part C

Proof-of-Work (PW): Implement a module for a miner to compute a PW where SHA3-224 is used as a hash function *h* in *PW₁* and *PW₂* computations.

1) **Find pre-images of *h* such that**

$$\boldsymbol{PW_1 = h(h(amt_0)||m_1||nonce_1) = 00 \ldots 0 ** \ldots *}$$
$$\boldsymbol{PW_2 = h(h(m_1)||m_2||nonce_2) = 00 \ldots 0 ** \ldots *}$$

where * means any value and $nonce_i$, i = 1, 2 are any 128-bit numbers. Here you use k = 32. You should vary a none in order to obtain a SHA3-224 hash value with 32-consecutive leading zeroes. Your results on hash values PW_1 and PW_2 should be represented as hexadecimal numbers.

Both $nonce_1$ and $nonce_2$ (can be found in the Appendix) were generated using the following brute-force procedure:

```python
def nonce(message):
  i = 0
  max_val = (2 ** 128) - 1
  while True:
    nonce = '{0:0128b}'.format(i)

    input = message + nonce

    # binary_string_to_hex first converts input to integer - causes loss of leading zeroes in returned hex string
    hex_input = binary_string_to_hex(input)

    # pad hex input with leading zeroes to account for loss; 1114 is the expected length of the hex string
    pad = 1114 - len(hex_input)
    padded_hex_input = ('0' * pad) + hex_input

    output = hex_to_binary_string(sha3_224_hex(padded_hex_input))
    if output[:32] == '{0:032b}'.format(0):
      return nonce
    if i == max_val:
      return None
    i += 1
```

The respective pre-images PW_1 and PW_2 (can be found in the appendix) were generated using the following function:

```
def construct_pre_image(arg1, arg2, nonce):
    return sha3_224_hex(binary_string_to_hex(hex_to_binary_string(arg1) + arg2 + nonce))
```

where $arg_1 = h(amt_0)$ and $arg_2 = m_1$ for PW_1, and $arg_1 = h(m_1)$ and $arg_2 = m_2$ for PW_2.

2) **Determine the average number of trials which you need to get one PW in (1).**

Given that the computed pre-image is 224 bits in length and the first 32 bits are set to zero, the remaining 192 bits can be any combination of zeroes and ones. As a result, there are 2^{192} possible pre-images with 32 leading zeroes out of a total 2^{224} combinations. Therefore, the probability of finding the correct pre-image with 32 leading zeroes is:

$$\frac{2^{192}}{2^{224}} = \frac{1}{2^{32}}$$