Q9. Hash. In this question, we aim to use SHA384 hash algorithm to find the nonce of the following magic sentence so that the hash has 6 leading zeros (24-bit zeros) in the beginning. Easy Python programming (less than 15 lines) will be needed in this question.

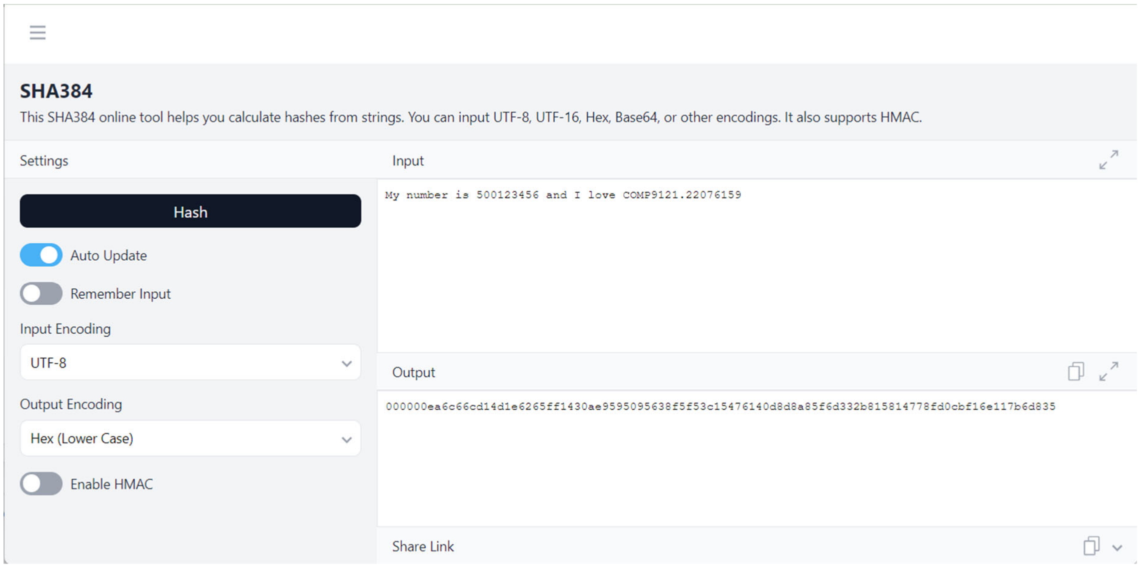
Magic sentence:

My number is ******** *and I love COMP9121.*########

Here ******** indicates your student number (so this should be different for everyone) and ######## is the "nonce" you need to find. ######## is a string, which can be with any length. For example, the magic sentence below is the outcome (the answer is not unique). Please note, there is no limitations on the length of the nonce.

My number is 500123456 and I love COMP9121.22076159

You can verify it at many websites, such as, https://emn178.github.io/online-tools/sha384.html, where you can see that the hash has 6 zeros in the beginning.



SHA384

This SHA384 online tool helps you calculate hashes from strings. You can input UTF-8, UTF-16, Hex, Base64, or other encodings. It also supports HMAC.

☑ Auto Update

☐ Remember Input

Input Encoding

UTF-8

Output Encoding

Hex (Lower Case)

☐ Enable HMAC

Input

```
My number is 500123456 and I love COMP9121.22076159
```

Output

```
000000ea6c66cd14d1e6265ff1430ae9595095638f5f53c15476140d8d8a85f6d332b815814778fd0cbf16e117b6d835
```

Please note that in the "input" in the above screenshot, there should be no carriage return or line feed. Otherwise, the hash will be different.

You need to use Python3 to find a nonce. (Since the answer is not unique, you only need to find one answer.) You may use hashlib and use a correct function in this library. For example, the following program will generate SHA384 of "*My number is 500123456 and I love COMP9121.*"

```
import hashlib
s="My number is 500123456 and I love COMP9121."
s=bytes(s,'utf-8')
hashresult=hashlib.sha384(s).hexdigest()
```

(1) Write down your magic sentence and submit Python 3 code as supplementary material.

(2) How long does it take to find the nonce?

(3) In this case we find the nonce generating for 6 leading zeros. Comment on what will happen if you are required to find a nonce that generates a hash with 20 leading zeros. (This is what miners of bitcoins should do!)