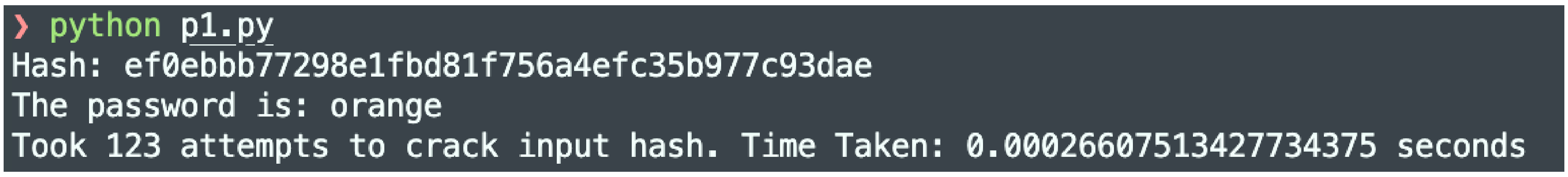
Quiz 2

Problem 1

In this problem, I used python package `requests v2.31.0` to fetch txt file of the given link. And I run `conda install requests` to install the package. Here is the plaintext of each hash:

1. Easy hash:

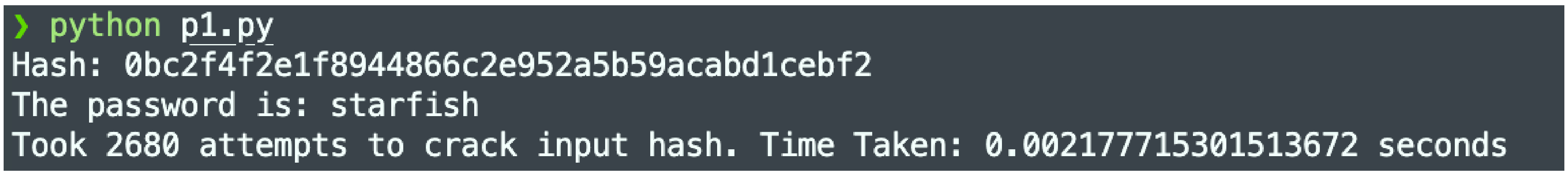
```
ef0ebbb77298e1fbd81f756a4efc35b977c93dae -> orange
```



```
❯ python p1.py
Hash: ef0ebbb77298e1fbd81f756a4efc35b977c93dae
The password is: orange
Took 123 attempts to crack input hash. Time Taken: 0.00026607513427734375 seconds
```

2. Medium hash:

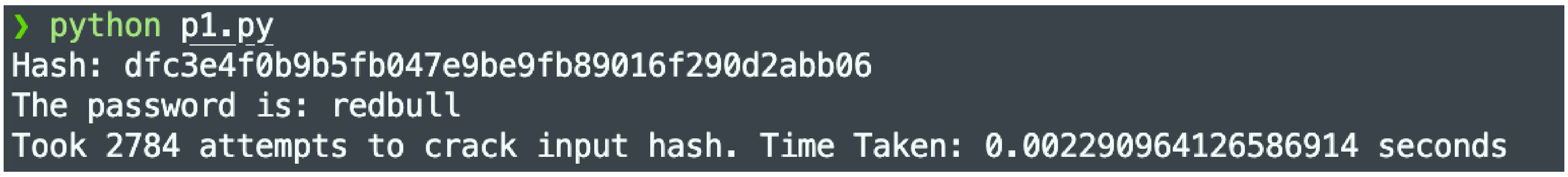
```
0bc2f4f2e1f8944866c2e952a5b59acabd1cebf2 -> starfish
```



```
❯ python p1.py
Hash: 0bc2f4f2e1f8944866c2e952a5b59acabd1cebf2
The password is: starfish
Took 2680 attempts to crack input hash. Time Taken: 0.002177715301513672 seconds
```

3. Leet hacker hash:
   First we decode the hash to **redbull**.

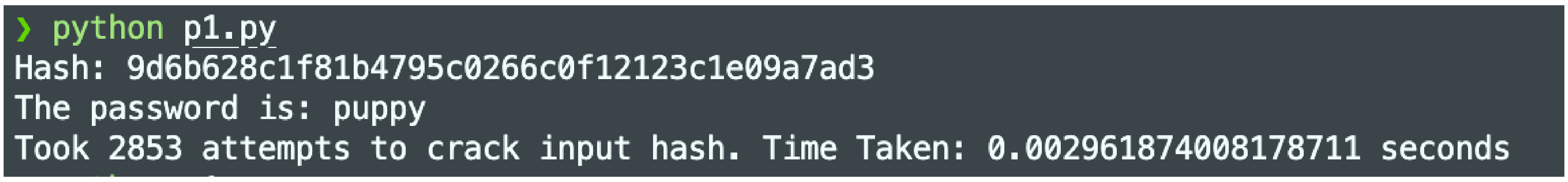
```
dfc3e4f0b9b5fb047e9be9fb89016f290d2abb06 -> redbull
```



```
❯ python p1.py
Hash: dfc3e4f0b9b5fb047e9be9fb89016f290d2abb06
The password is: redbull
Took 2784 attempts to crack input hash. Time Taken: 0.002290964126586914 seconds
```

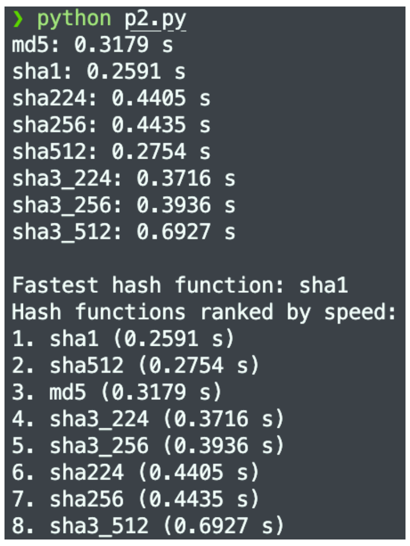
Then we can use redbull and another word to match the hash.

```
9d6b628c1f81b4795c0266c0f12123c1e09a7ad3 -> rebull + puppy
```



```
❯ python p1.py
Hash: 9d6b628c1f81b4795c0266c0f12123c1e09a7ad3
The password is: puppy
Took 2853 attempts to crack input hash. Time Taken: 0.002961874008178711 seconds
```

Problem 2



```
❯ python p2.py
md5: 0.3179 s
sha1: 0.2591 s
sha224: 0.4405 s
sha256: 0.4435 s
sha512: 0.2754 s
sha3_224: 0.3716 s
sha3_256: 0.3936 s
sha3_512: 0.6927 s

Fastest hash function: sha1
Hash functions ranked by speed:
1. sha1 (0.2591 s)
2. sha512 (0.2754 s)
3. md5 (0.3179 s)
4. sha3_224 (0.3716 s)
5. sha3_256 (0.3936 s)
6. sha224 (0.4405 s)
7. sha256 (0.4435 s)
8. sha3_512 (0.6927 s)
```

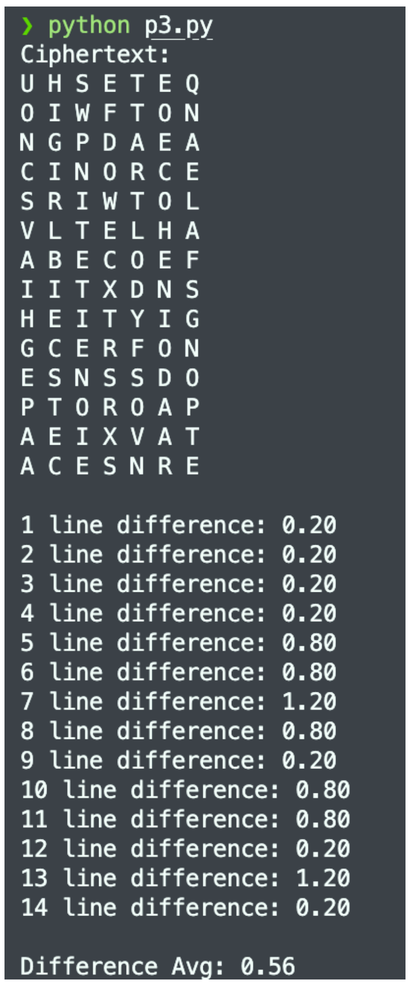
As the picture shown above, we can know the fastest algorithm is "sha1". And the rank of each algorithm is:

1. SHA1

2. SHA512
3. MD5
4. SHA3-224
5. SHA3-256
6. SHA224
7. SHA256
8. SHA3-512

Problem 3

- Avg difference of 14 x 7 rectangle = 0.56:

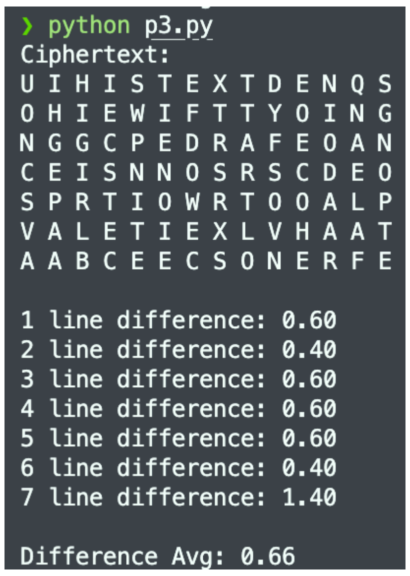


```
❯ python p3.py
Ciphertext:
U H S E T E Q
O I W F T O N
N G P D A E A
C I N O R C E
S R I W T O L
V L T E L H A
A B E C O E F
I I T X D N S
H E I T Y I G
G C E R F O N
E S N S S D O
P T O R O A P
A E I X V A T
A C E S N R E

1 line difference: 0.20
2 line difference: 0.20
3 line difference: 0.20
4 line difference: 0.20
5 line difference: 0.80
6 line difference: 0.80
7 line difference: 1.20
8 line difference: 0.80
9 line difference: 0.20
10 line difference: 0.80
11 line difference: 0.80
12 line difference: 0.20
13 line difference: 1.20
14 line difference: 0.20

Difference Avg: 0.56
```

- Avg difference of 7 x 14 rectangle = 0.66:

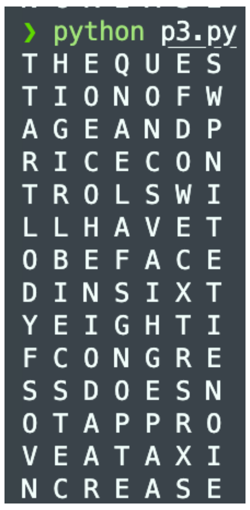


```
❯ python p3.py
Ciphertext:
U I H I S T E X T D E N Q S
O H I E W I F T T Y O I N G
N G G C P E D R A F E O A N
C E I S N N O S R S C D E O
S P R T I O W R T O O A L P
V A L E T I E X L V H A A T
A A B C E E C S O N E R F E

1 line difference: 0.60
2 line difference: 0.40
3 line difference: 0.60
4 line difference: 0.60
5 line difference: 0.60
6 line difference: 0.40
7 line difference: 1.40

Difference Avg: 0.66
```

So we know 14 x 7 is better size than 7 x 14. Next we can try to shift columns according to ***"the"*** word, then we could guess ***"ques"*** may be the following four character in right position, which is the answer.



```
❯ python p3.py
T H E Q U E S
T I O N O F W
A G E A N D P
R I C E C O N
T R O L S W I
L L H A V E T
O B E F A C E
D I N S I X T
Y E I G H T I
F C O N G R E
S S D O E S N
O T A P P R O
V E A T A X I
N C R E A S E
```