Glenn Chia 1003118 (Lab 3 section 6)

1. Attempted approaches

I knew that generating rainbow tables would be infeasible as question 6 could have original strings of different length and any character/symbol/number. Hence, I took the following approaches

1. Using a large text corpus of common passwords, hashing their values and using it as a dictionary to compare with
2. Using an online API that has been trained on
3. Combining the above Approach 1 and Approach 2

2. Approach 1: Dictionary only

I downloaded a large corpus from https://github.com/danielmiessler/SecLists/blob/master/Passwords/Common-Credentials/10-million-password-list-top-1000000.txt

- I then ran some code to iterate through the text file, hash each value and create a dictionary with hash as the key and original string as the value
- I then iterated through hashes.txt and tried to find mappings
- This produced a score of **25%** - not good enough
- I knew that doing a rules based approach (e.g. swapping a with @ etc.) would not be effective as the text corpus has situations for p@ssword and password for instance among others

3. Approach 2: Using an online API

I went to the following link `https://www.nitrxgen.net/md5db/<hash_here>.json` which was an online website with 1.1 Trillion passwords

This approach was significantly better and achieved **142/148** passwords

4. Approach 3: Approach 2 + Creating my own dictionary

For the remaining passwords, I checked on various online websites to crack it. Only one website worked https://hashkiller.co.uk/Cracker/MD5. Hence I created a mapping for those hashes and their original values and used this as the dictionary.

I was then left with these unidentified hashes. Score **145/148**

- 4698e7ab9c06649f06f3bbc8fcb20360
- daac6467d1d7cb418572dbd8d01c190a
- d768d3b271ba9faaab0141600a47b221F

5. Conclusion

After looking at the original texts, some of them seem completely random and not related to any words at all or were a concatenation of words. There was no way to crack them using a rules based approach. If possible, always use tools and large databases to crack the hashes as these are the most robust platforms and the fastest, most cost-efficient way to crack.