CS 3339
LAB 3 - Password Cracking

You will have 2 weeks to do this lab.

In this lab we will be cracking passwords. You will be given a list of password digests (**hashes.txt**) and you must determine the corresponding plain text password for each hash in this file. All of the hashes in this lab were hashed using SHA256. All passwords are salted with the randomly generated salt found in **salt.txt** . In python, the salt is applied and the message is hashed like this:

hashlib.sha256((salt + password).encode())

For a more complete example, look at the included code in **hash.py**. I used this exact code to create **hashes.txt** from a password list. I also included an additional example that you can try with this program (**example_passwords.txt** and **example_hashes.txt**)

In addition, you may **NOT** use any password cracking tool to do this lab.
You must write your own password cracker to use.
It can be done in whatever language you feel comfortable with (I suggest Python)

You should at least implement a brute-force and a dictionary attack.
Additional attack types may be necessary to find all passwords

Types of passwords to look for

- English words and names (first letter capitalized, no capitalization, or all capitalized)
    - Eggplant, waterfall, KANGAROO
- Long english words (11-26 characters)
- 2 common words concatenated (no spaces, no capitalization)
- Common passwords that I pulled from the Internet
    - Find a large common password list to parse
- Random strings (up to 4 characters in lowercase letters, numbers)
- Common English words with trailing numbers and symbols (up to 3) (not capitalized)
    - nerd12!
    - Symbols may include !@#$%^&*-+=<>?
- English words with all vowels replaced with similar symbols and numbers
    - l33t h@ck0r

Deliverables

- Your password cracking program source code
- A README for compiling/running your program
- A file named cracked.txt which should contain all of the passwords that you have found and their corresponding hashes in the following format. If a password is not found, just include the hash followed by a blank.

    <hash>:<password>
    <hash>:

    All deliverables should be uploaded to canvas in a .zip file

    Grade: 50 points for the program, 55 points for passwords (1 point per password)