Kaitlyn Chau
Professor Rahmati
CSE 331
10/2/2024
Homework 2

How to run the code:
$ ./taskN.sh password_database.csv

Task 1:
Read the input password database csv file line by line, extracting the tokens. When begin looking at the hashed unsalted passwords, make a for loop of only the pw lengths we are considering (no more than 4 chars) and take note of time. Take the cartesian of all the possible alphanum passwords of the length we are looping at and hash it to see if it matches the data entry. Finish by writing to the corresponding csv file.

Task 2:
Note that the common_passwords.csv file from the linked kaggle has to be included in the same directory. I included that file in the tar zip in case. To run the code, include the input password_database.csv in the cmd line as an argument such as $./task2.sh password_database.csv . The code takes in the path for the pw_db.csv and extracts the lines as tokens. Process the common passwords and make it so that there is a dictionary of the hashed password to the plaintext password. When we actually start looking at the input passwords, take note of start time. Loop through the data rows and check if the hashed unsalted password is in the dictionary. If it is or not, add to the array of output messages. Finish by writing to corresponding csv file.

Task 3:
The code takes in the path for the pw_db.csv and extracts the lines as tokens. Process the common passwords and make it so that there is a dictionary of the hashed password to the plaintext password. When we actually start looking at the input passwords, take note of start time. Loop through the data rows and check if the hashed unsalted password is in the dictionary. If it is or not, add to the array of output messages. Finish by writing to corresponding csv file.

Task 4:
The code takes in the path for the pw_db.csv and extracts the lines as tokens. Process the common passwords and *salt it* and make it so that there is a dictionary of the hashed password to the plaintext password. When we actually start looking at the input passwords, take note of start time. Loop through the data rows and check if the hashed unsalted password is in the dictionary. If it is or not, add to the array of output messages. Finish by writing to corresponding csv file.

Reference:

https://www.kaggle.com/datasets/shivamb/10000-most-common-passwords?resource=download