

**İSTANBUL ÜNİVERSİTESİ-CERRAHPAŞA**

**COMPUTER ENGINEERING**

**FILE ORGANIZATION**

TERM PROJECT

NAME : MUHAMMET TALHA ODABAŞI

**ID**  : 1306220012

PROGRAMMING LANGUAGE : PYTHON

1. INTRODUCTION

In my project, I carried out all operations with the OS and hashlib library using the Python programming language. The goal of the project is, as stated in the document shared on google classroom, using password lists on SecList to collect and process raw passwords to create an Index file and to facilitate the search process

1. USED TECHNOLOGIES

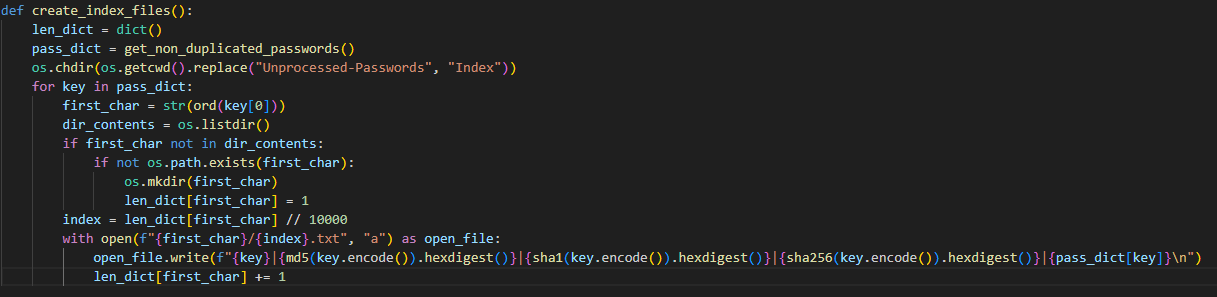
As I explained in the introduction, I wrote the code using Python 3.11 and libraries I used are:

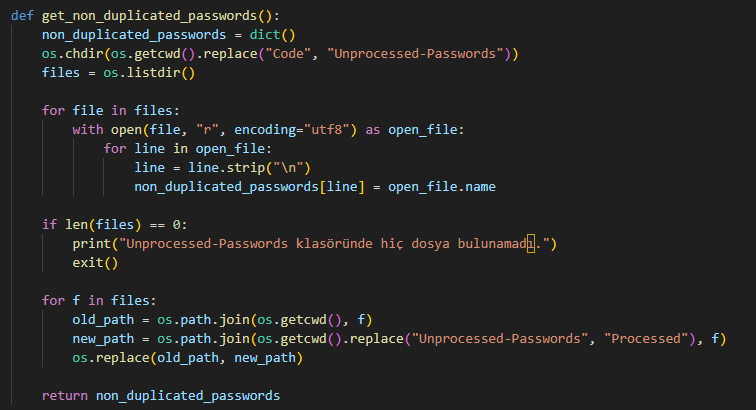
* I performed file operations using OS library, used functions such as getcwd(), listdir()
* I calculated MD5, SHA1, SHA256 hash values with the hashlib library.
* I calculated how long it took to find the password with the datetime library.

1. Project DESIGN

* There are unprocessed password files which are taken from SecLists repository in Unprocessed-Passwords directory.
* Processed directory is used to store password files that are processed. After processing a file it is moved from Unprocessed-Passwords to Processed.
* Index directory holds indexed versions of all passwords. It creates folders with ascii number value of password’s first character and stores passwords in a file inside that directory such as 0.txt which holds 10000 records.
* Code directory includes (main.py) program code.

1. Indexıng



****

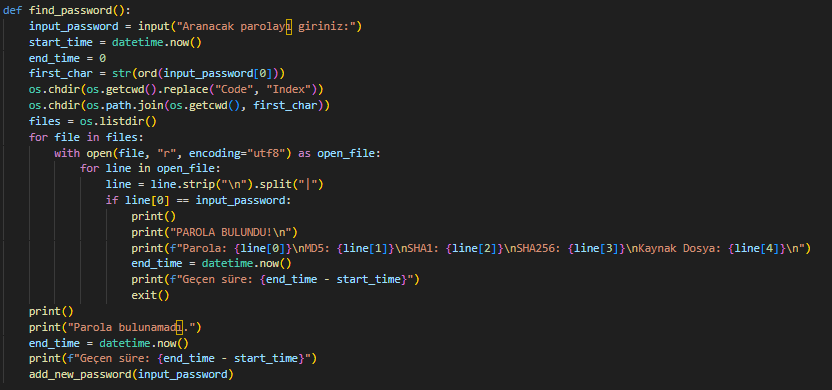
As seen on above codes, all files inside Unprocessed-Passwords are looped line by line and stored inside a dictionary with key of password and value of source file. It is stored in password to make sure no duplicate passwords are processed.

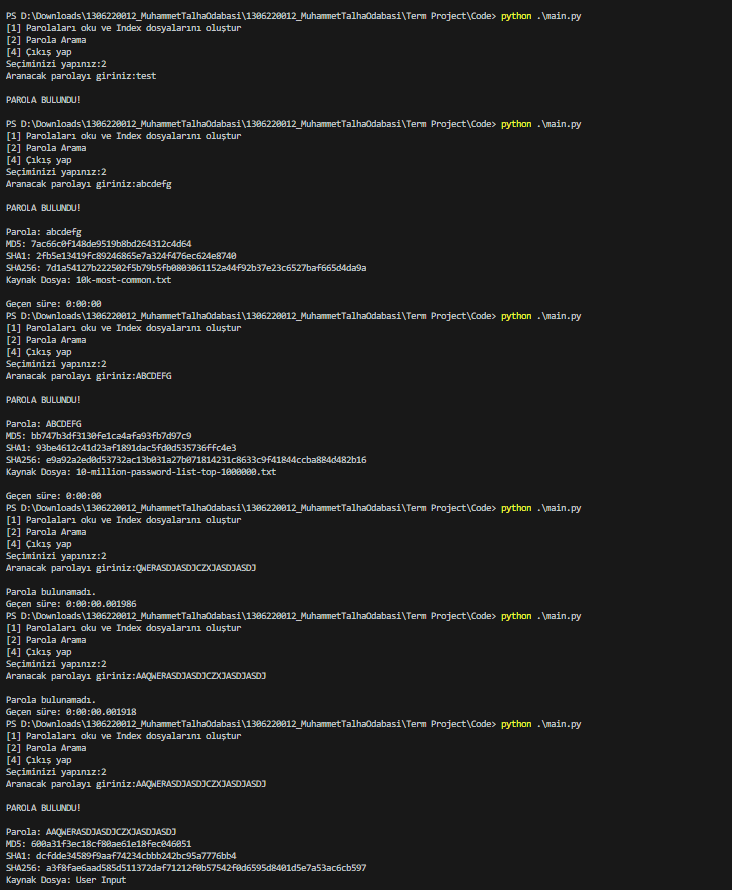
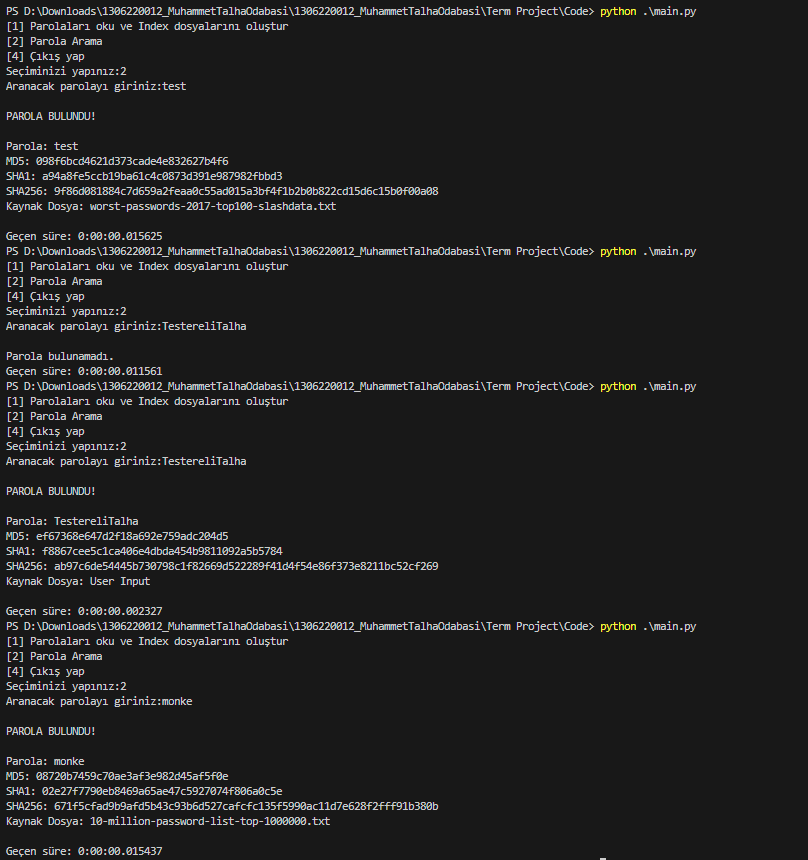
After that dictionary is looped and all passwords are stored with specified format inside Index folder. Ascii numbers are used as folder names to be able to create folders of special characters. Moreover, len\_dict is used to be able to store files up to 10000 lines and then seperate.

****

This is the time it takes to index all files.

1. SEARCH FUNCTION

This is the search function. It gets a password input and searchs for the password inside Index folder. If password is not found it adds it to Index folder with source as User Input.



These are from performance tests. It takes microseconds to find or not find a password. And as you can see, If password is not found on first try it is found on second one.

1. CONCLUSION

As a result of indexing, the search process has visibly accelerated. Considering that this is our purpose I can say that we fulfilled our purpose. Doing the project with C style low level languages would take longer but its indexing and search ability would be faster. Moreover, this project can be used to create mini relational database.

1. Source

I did not use any specific source, I just used stackoverflow and GPT 3.5 to create specific algorithms.