Ashley Cannon

CS110

Prof. Ryan

Overview and Summary of Project

The main goal of this project is to encrypt and decrypt passwords within the same program. The program prompts the user to enter a password, then outputs the encrypted password as well as the decrypted password. There are multiple functions involved in achieving this output, the encryption function as well as the decryption function.

Text

Description automatically generated

Text

Description automatically generated

Target Audience

The target audience for this program would be anyone who want extra protection beyond their password. Passwords should always be strong, but some people want extra protection on higher risk accounts such as bank accounts, logins for their job, and anything else that may be important to them. The ability to encrypt a password makes it much safer for the sensitive accounts previously mentioned. It can be easy for hackers to access such accounts if there is a simple password in place, however the use of an encrypted program will ensure that the passwords are protected.

Specific Programming Techniques Used

This program uses a large variation of techniques such as functions, lists, for loops, and importing the math library. The math library is imported at the top of the function to be used later on. Initially, the user is prompted to input the password to be encrypted. After the encrypt function has been called, the program then defines an empty list for the encrypted characters to go in. The characters in the password are then changed into ASCII values by using the ord function, which allows for the manipulation of values. These values go through a for loop and into the empty list. Append is then used to put all the new values into the list. The values of the encrypted password are printed with the message, ‘This is your encrypted password’. After the function has been encrypted, the decrypt function is called to return the values back to the original input. Another for loop is used to undo the encryption done in the previous for loop. The program then prints the resulting output, which should be the original password.

Challenges

There were only a few minor challenges presented when writing this program. One of the biggest was trying to determine the level of difficulty of the algorithm. The algorithm used to encrypt the passwords is relatively simple, because the more complicated it was, the harder it was to decrypt it. For my level of programming, this was difficult enough to decrypt correctly. There were many different error messages while trying to get the program to function the way I wanted it to. The best way to overcome these was through trial and error. Fixing one mistake at a time allowed for successful use of the program.

Future Extension

A future extension of this program could be to design a new part of the program that tested the strength of the password. An algorithm could be created that determined whether or not the password being used could be hacked, and suggest a new password based on the encryption of the old password. For example, if the password was simply password, it could encrypt the characters, and instead of decrypting, create a new password based on the characters of the new values. There would need to be restrictions on the numerical values used, but it could create the possibility of stronger passwords.

Text

Description automatically generated