Hola a todos, mi nombre es Luis y les doy la bienvenida a Mi Diario Python. En el día de hoy les mostrare un ejemplo sencillo de cómo hacer un algoritmo de fuerza bruta para mensajes cifrados con el cifrado cesar.

En el artículo anterior vimos un ejemplo sobre un algoritmo que cifraba y descifraba mensajes con el Cifrado Cesar. Puedes ver el artículo ingresando al siguiente enlace:

Como sabemos, para cifrar un mensaje, debemos elegir una key, la cual será como la combinación del cifrado. Al momento de cifrar podemos ingresar la key que deseamos con respecto a la longitud del mensaje. Para descifrar un mensaje debemos de saber la llave, lo que nos lleva a ir probando llave por llave. Para evitar esto, y ahorrar tiempo, escribiremos un algoritmo que pruebe descifrar un mensaje utilizando distintas llaves de forma automática.

¿Qué es fuerza bruta? Hace un tiempo atrás publique un artículo en donde explicaba con un ejemplo la fuerza bruta a cuentas de gmail. Puedes ver ese artículo ingresando al siguiente enlace:

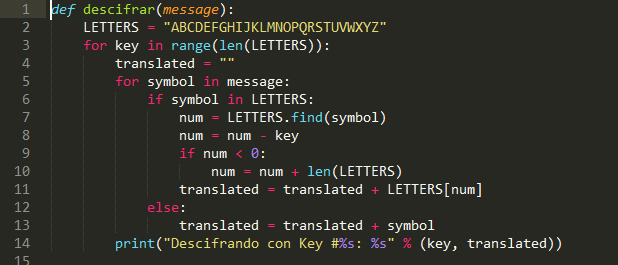
Perfecto, una vez que conozcamos el concepto de Cifrado Cesar y Fuerza Bruta, estamos listos para escribir un algoritmo que los combine a los dos.

Algoritmo: Fuerza Bruta – Cifrado Cesar.

Lo que hará el algoritmo, será tratar de descifrar un mensaje cifrado probando 21 llaves diferentes. Todo de manera automática, de manera que nos ahorre el tiempo de ir probando llave por llave.

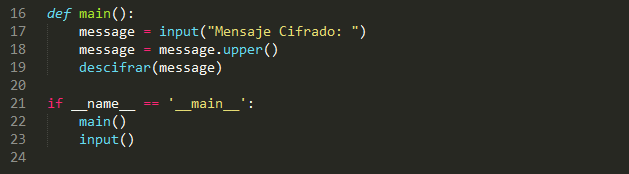
Comencemos abriendo nuestro editor de texto preferido. En mi caso utilizare sublime text.

Empezaremos escribiendo la función “descifrar” la cual hará todo el procedimiento de descifrado.



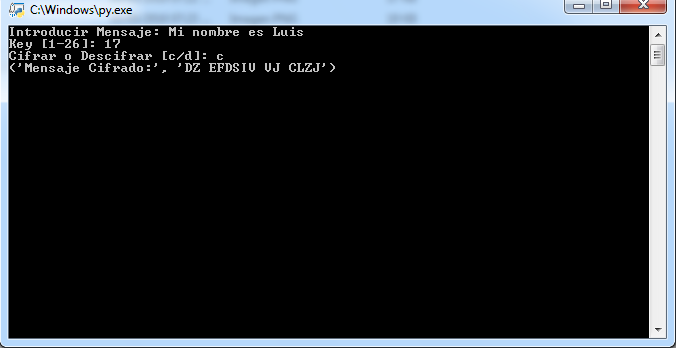
La función recibe como argumento el mensaje cifrado. Declaramos una variable LETTERS la cual contendrá todas las letras del abecedario.

Luego escribimos un bucle for el cual no se detendrá hasta las 26 iteraciones. Como se puede observar, este será el número de key que se probara para descifrar el mensaje. AL final se mostraran los distintos resultados.



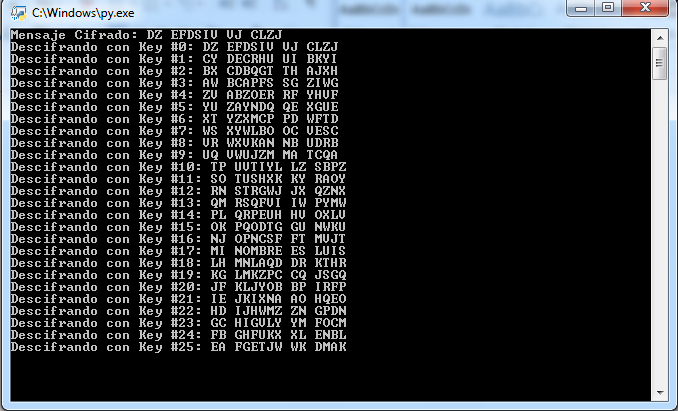
Por último, escribimos la función la “main” la cual pedirá el mensaje cifrado al usuario para luego ser almacenado en la variable message. La función upper() convertirá todo al texto a mayúsculas, así evitaremos errores al trabajar con las cadenas de texto.

Listo, ya podemos ejecutar el script e ingresar el mensaje cifrado en mi caso, utilizare un ejemplo realizado en el artículo anterior:



Ese será el mensaje cifrado que utilizaremos en el día de hoy. Observen que la key utilizada es 17.

Ahora sí podemos ejecutar el script de Fuerza bruta:



Como pueden observar, al ingresar el mensaje cifrado me aparecen las distintas combinaciones.

Si pueden observar, en la llave #17 se muestra “MI NOMBRE ES LUIS”. Este resultado es correcto.

Así que el algoritmo cumplió su objetivo de vida.

¿Qué te pareció? ¿Alguna duda? No olvides dejar un comentario.

Mi nombre es Luis, y fue un placer compartir mis conocimientos con todos ustedes :D.