Reporte del proyecto de criptografía

Dificultades encontradas

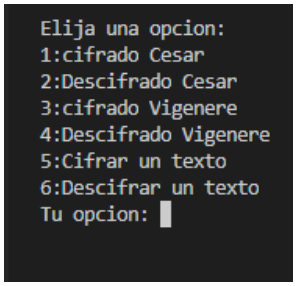
Desde el inicio del proyecto nos quedó bastante claro el flujo que debía tener el programa, así como el algoritmo de cifrado y su implementación; sin embargo sí tuvimos una dificultad difícil de sortear. La no unicidad del código ASCII[1] extendido como norma de codificación, dificultó el proceso de pasar de los 127 primeros caracteres universales[2] al ASCII extendido "tradicional"[3], sobre todo al tratar de escribir en español o francés.

Soluciones

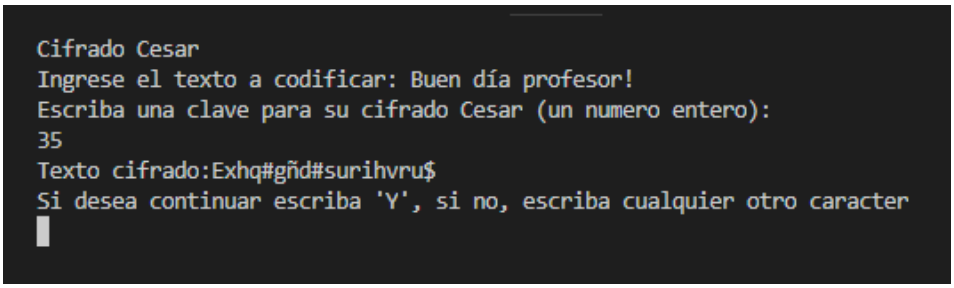
Para solucionar este problema, se pensó utilizar UNICODE debido a su universalidad. Sin embargo, al ser más difícil trabajar con números en hexadecimal, y el hecho que algunos caracteres en UNICODE requieren hasta 6 bytes para ser almacenados, se descartó esta opción. Por lo tanto, el programa se hizo pensando en que la mayoría de las computadoras modernas de occidente, utilizan la norma ISO 8859-1 de codificación de caracteres.

Características y opciones del programa

El programa contiene 6 opciones,



1- Cifrado César

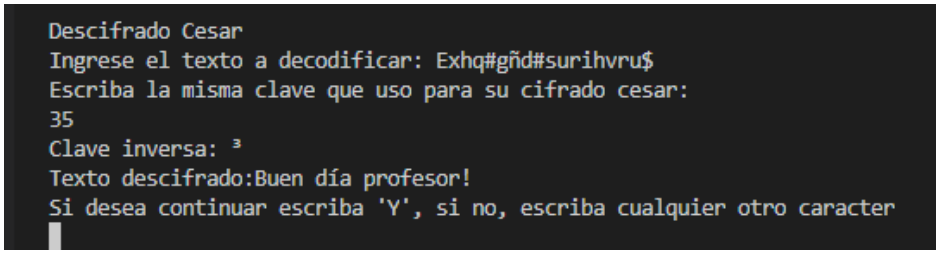


[1] ASCII es la sigla de: American Standard Code for Information Interchange.

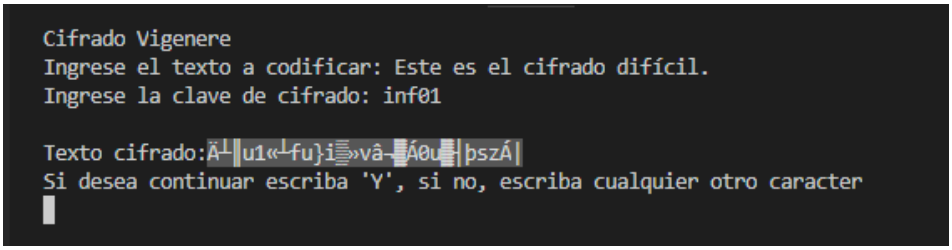
[2] Exceptuando los 31 primeros correspondientes a los caracteres de control.

[3] La versión del ASCII correspondiente a Windows-1252, que es una copia del ISO 8859-1 en cuanto a caracteres imprimibles se refiere.

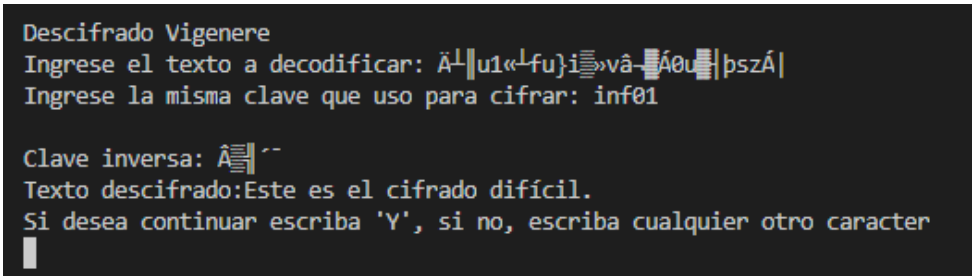
2- Descifrado César



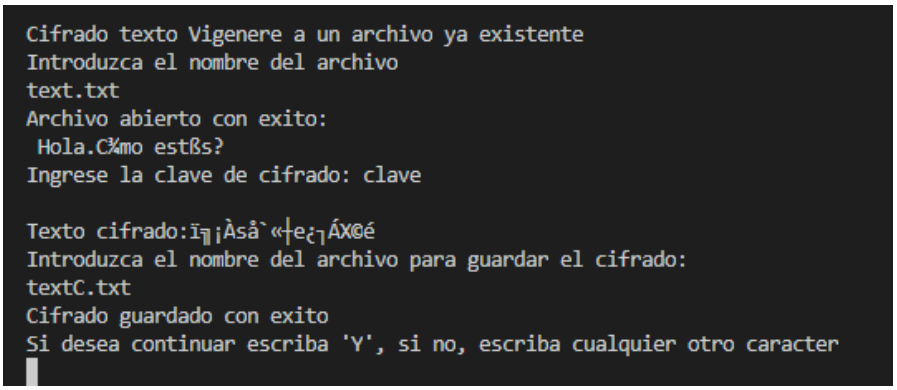
3- Cifrado Vigènere(introduciendo un texto en la consola)



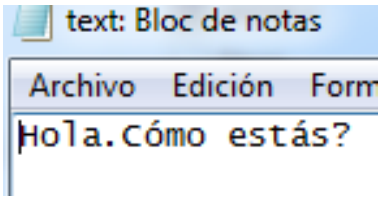
4- Descifrado Vigènere(introduciendo el texto cifrado en la consola)



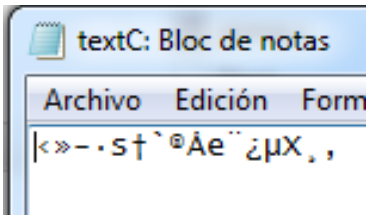
5- Cifrado Vigènere (leyendo un archivo de texto ya existente)



Texto leído: text.txt

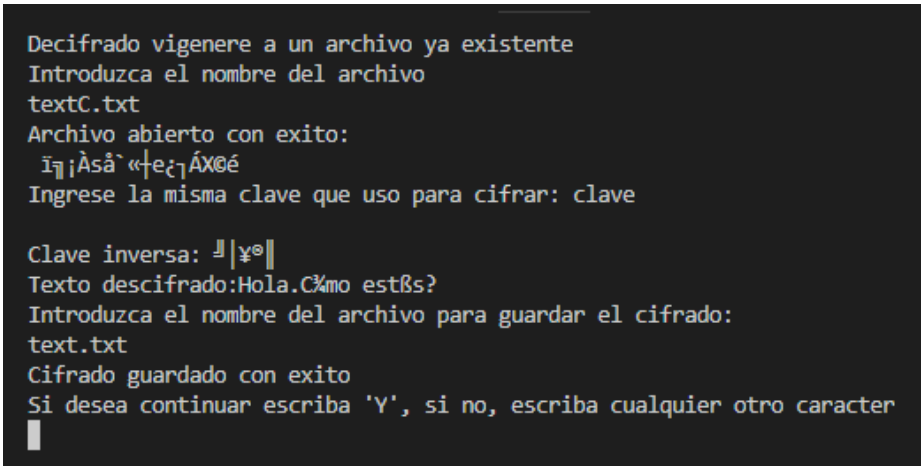


Texto escrito: textC.txt

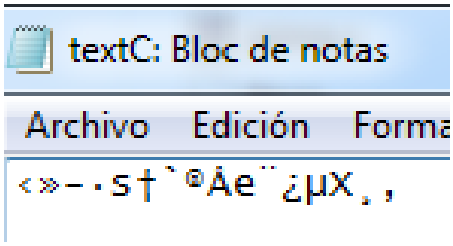


Como podemos ver, hay varias incongruencias. La primera de ellas, es la forma en que se lee el archivo en el programa, sustituyen los caracteres con acento por otros diferentes. La segunda la encontramos cuando comparamos el texto guardado en el archivo textC.txt y el texto cifrado desplegado en consola. Sin embargo, procederemos a descifrar el archivo text.txt.

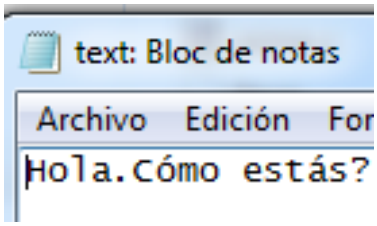
6- Cifrado Vigènere (leyendo un archivo de texto ya existente)



Texto leído: textC.txt



Texto escrito: text.txt



Como podemos notar, las incongruencias entre los textos mostrados en consola y en los archivos de texto permanece, pero las incongruencias son consistentes.

¿Qué pasó?

Todos los caracteres están codificados en tablas que corresponden a un número. En el caso de esta versión de C y el compilador GCC utilizado, la codificación para los caracteres en consola es la ASCII, particularmente la versión que IBM publicó en 1981[4], conocida como "code page 437", que se conoce también como ASCII extendido. Sin embargo, la codificación estándar de archivos .txt es de acuerdo a la tabla ANSI[5], por lo que hay una diferencia en la representación de los caracteres para un mismo número.

Es por eso que el carácter "á" guardado en el archivo es representado como "ß" en la consola; ya que "á" corresponde al número 225 en el codificado ANSI, y el 225 en ASCII extendido es precisamente "ß". Hay además, otra cosa que podemos notar, y es que no todos los caracteres de la cadena mostrada en consola difiere de la cadena guardada, y esto es porque el ASCII tradicional de 7 bits es congruente con todos los sistemas de codificación, por lo que cualquier carácter contenido entre 32 y 127 se representa de la misma forma en consola y en el archivo de texto (en ASCII y en ANSI). Ahora bien, el programa funciona, porque el algoritmo de cifrado está basado en números asociados a caracteres, y el número en cuestión no cambia, lo que cambia es su representación.

Es imposible convertir el ASCII extendido en ANSI y viceversa, ya que, a pesar de que ambos ocupan 8 bits, los caracteres asociados a ASCII son diferentes a los de ANSI, por lo que aun haciendo un diccionario de datos, sería imposible hallar una relación 1:1 entre los caracteres de ASCII y ANSI. Para ejemplificar lo anterior, el caracter "€" que representa el número 128 en la tabla ANSI, no existe en la tabla ASCII. Para solucionar este problema, se inventó UNICODE, que es la norma internacional de codificación de caracteres, y abarca más de 1 millón de caracteres.

[4] Para ver la tabla ASCII extendida: https://theasciicode.com.ar/
[5] Para ver la table ANSI: http://ascii-table.com/ansi-codes.php