100cias@uned
Facultad de Ciencias

N.º 3 (nueva época) | 2010
ISSN: 1989-7189

Enseñanza

TALLER Y LABORATORIO

LABORATORIO DE MATEMÁTICAS

CIFRADOS CESAR ALEATORIOS: LA ENCRIPTACIÓN POR SUSTITUCIÓN DESDE LOS NÚMEROS RACIONALES A LOS IRRACIONALES

Al leer este texto nos encontramos con diversas agrupaciones de letras que nos resultan familiares y a las que asignamos significados, es decir, leemos y reconocemos las distintas palabras de nuestro lenguaje. Como toda lectura de un artículo, ésta se inicia con la lectura del título, que por simplificar lo escribimos como: "CIFRADOS CESAR ALEATORIOS", constituido por tres palabras del lenguaje español.

¿Qué interés hubiera despertado al lector encontrar un título como: "FLIUDGRV FHVDU DÑHDWRULRV"?

Al parecer, nadie escribe un título de este tipo. Un título como este último es difícil de leer y de recordar, mucho más difícil resulta reescribirlo correctamente sin poder verlo. Puede parecer que este último título está compuesto por otras tres palabras en algún lenguaje desconocido, esto es debido a que estamos acostumbrados a distinguir las palabras como aquellas agrupaciones alfabéticas contiguas que están delimitadas por dos símbolos no alfabéticos.

¿Quién querría titular sus artículos con expresiones textuales como las del último título, "sin significados y que no tienen interés alguno"? Probablemente nadie, pues precisamente se suelen escoger títulos que comuniquen la esencia del contenido del artículo.

¿Cómo podemos convencer al lector de que esos dos títulos anteriores tienen el mismo significado? Espero que la lectura de este artículo dé respuesta a esta pregunta.

Amigo lector, si por casualidad un título como "FLIUDGRV FHVDU DÑHDWRULRV" no le hubiese despertado ningún interés, entonces habría conseguido aquello que deseaba Cayo Julio Cesar cuando enviaba mensajeros portando epístolas conteniendo textos similares. Textos en los cuales describía algunos planes secretos o indicaba aquella acción que debían ejecutar los receptores. Su intención última era mantener informado a sus generales de manera que si el mensajero o el mensaje cayera en manos enemigas, éstos no pudieran comprender lo escrito.

No queremos emular a Cesar, es decir, no queremos ocultar nada de lo escribimos. Simplemente queremos aportar algunas técnicas de encriptado de mensajes y documentos desde un punto de vista divulgativo. Para ello acompañamos el texto con un mini-laboratorio de cifrado-descifrado, y el lector podrá realizar algunas prácticas sencillas con él. Este laboratorio está programado en el lenguaje de Maple. En su momento, descartamos utilizar algún programa de C^{++} o de Java con el cual se pudiera interactuar, pues la posibilidad de que el lector analizara la pila de instrucciones Maple y las modificara a su antojo, nos parecía más sugestiva. El código Maple no es muy extenso y puede ser teclearlo en poco tiempo sin los comentarios. Si lo desea puede pedir este código al autor utilizando el correo electrónico.

Este trabajo se inicia en una breve conversación con el profesor Dr. J. Carlos Antoranz que duró lo que duraba el compartir un "sucedáneo de café" erogado por la antigua máquina de la Facultad de Ciencias (UNED). Sin duda, este encuentro fue una nueva utilidad de esa máquina.

Este profesor me proponía crear un mensaje cifrado, el decía tipo Cesar, donde la clave fuera variable, obteniéndola recursivamente empleando una función que mediante iteración presentara una situación caótica.

En este trabajo no se pretende hacer criptoanálisis de lo que presentamos, quizás ésa sea la misión del lector interesado, o de algún futuro trabajo.

1. INTRODUCCIÓN

Al intentar plasmar el lenguaje hablado se hace uso de diversos símbolos para poder dibujar, negro sobre blanco, los fonemas que empleamos, su ritmo de aparición y sus pausas. Con este proceso se trata de almacenar los conocimientos orales deseados de una forma duradera. De esta forma, una vez almacenados, el conocimiento puede ser compartido, replicado y enviado de un sitio a otro y ser accesible por mucho tiempo.

Un problema se nos plantea si queremos controlar el acceso a dichos conocimientos. Sin duda, se pueden diseñar estrategias mixtas según un reparto porcentual de las dos estrategias básicas siguientes:

- Custodiar la base de conocimiento, restringiendo el número de personas que pueden acceder a ella.
- Crear una base de conocimientos grande que envuelva al conocimiento base que queremos almacenar como una parte insignificante y, de esa forma, ocultar su significado.

En el lenguaje escrito se emplean: un juego de símbolos o alfabeto, unas reglas gramaticales para combinar esos símbolos y determinados valores semánticos de algunas agrupaciones de símbolos (palabras). Si no se sabe leer, los significados del escrito quedan ocultos. Si se sabe leer, entonces deben emplearse técnicas externas al lenguaje para ocultar significados.

En algunos grupos sociales emerge de forma ingenua la forma de controlar el acceso al conocimiento haciendo que las palabras presentes en un escrito pueden ser utilizadas con significados distintos a los tradicionales. Este secretismo es fácil de romper y sólo requiere cierto periodo de aprendizaje para poder entender este tipo de texto, aunque puede tener un coste personal más o menos grande. Ahora bien, no cabe duda de que no hace falta aprender nuevas palabras, pues se emplea un lenguaje natural. Sólo se necesita aprender la nueva asignación de significados para cada palabra. El conocimiento se mantiene secreto hasta que aparece un delator.

Para ocultar el conocimiento es mejor estrategia cambiar algunas palabras por otras agrupaciones alfabéticas. Como paradigma está el caso de cambiar de idioma, como hacían el grupo de telegrafistas formado por indios navajos en la segunda guerra mundial. El conocimiento se mantiene secreto hasta que se aprende ese idioma.

Sin duda es preferible que las nuevas agrupaciones textuales no estén registradas como palabras de ningún idioma. Eso era lo que hacía Cesar.

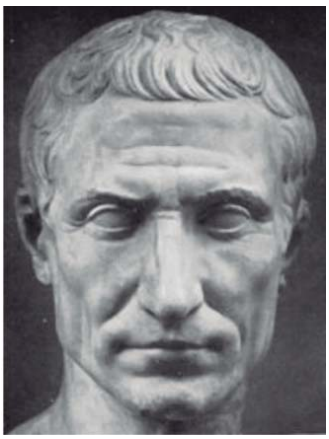

Figura 1. Dibujo de una estatua de Cayo Julio Cesar.

Cayo Julio César, como militar y político hace más de 2000 años, necesitaba asegurarse que aquello que escribiera a sus generales y colaboradores no fuese interpretado por sus oponentes o enemigos. Empleó, como en cada época ocurre, una estrategia acorde al desarrollo tecnológico de la época, sencilla y exitosa, que es denominada *cifrado César*. Hoy empleamos ordenadores para mantener nuestras estrategias factibles.

¿En qué consiste este sistema estratégico? Se trata de crear un texto no legible, "FLIUDGRV FHVDU DÑHDWRULRV", denominado *texto cifrado* o *encriptado*, tras realizar una transformación sencilla en cada uno de los caracteres de un texto legible, "CIFRADOS CESAR ALEATORIOS" que se denomina *texto claro*. Cesar escribía sin espacios en blanco para no informar sobre del número de palabras, es decir:

FLIUDGRVFHVDUDÑHDWRULRV y CIFRADOSCESARALEATORIOS

El cifrado más utilizado por Cesar transforma cada letra del texto claro en la letra que está tres lugares más adelante en el abecedario, suponiendo que la letra siguiente a *Z* es la letra *A*.

Este cifrado pertenece a los llamados *cifrados por desplazamiento*, caracterizados por reemplazar cada letra del texto original por otra que se encuentra un número fijo de posiciones más adelante en el alfabeto.

En el ejemplo no ha se utilizado el alfabeto latino, se ha utilizado un alfabeto constituido únicamente por las veintisiete letras mayúsculas del alfabeto español actual. Sin embargo, en el trabajo práctico utilizamos un alfabeto constituido sólo por números, concretamente el intervalo de los números naturales $[0,26\,]_{\mathbb{N}} = \{0,1,2,3...26\}$. Este cambio siempre es posible, pues entre cualquier alfabeto de símbolos y un intervalo de números similar al anterior existe una aplicación biyectiva (aplicación de codificación, decodificación) que asigna a cada símbolo un único número. Dado un alfabeto Ω' con $|\Omega'|<\infty$, siendo $|\Omega'|$ el número de elementos de Ω', se puede establecer una biyección con el conjunto de menores residuos módulo $|\Omega'|$, $\Omega=[0,|\Omega'|-1]_{\mathbb{N}}$.

Aunque en los ejemplos sólo se utiliza un juego de símbolos compuesto por las veintisiete letras mayúsculas del lenguaje español (26 serían en inglés), se puede utilizar cualquier otro alfabeto. El juego de símbolos puede ser pequeño como el conjunto de bits distintos, {0,1}, mediados como los 128 símbolos ASCII estándar o los 256 del ASCII extendido, o grandes como el conjunto de símbolos UNICODE, el cual engloba casi todos los símbolos presentes en los principales lenguajes escritos del mundo.

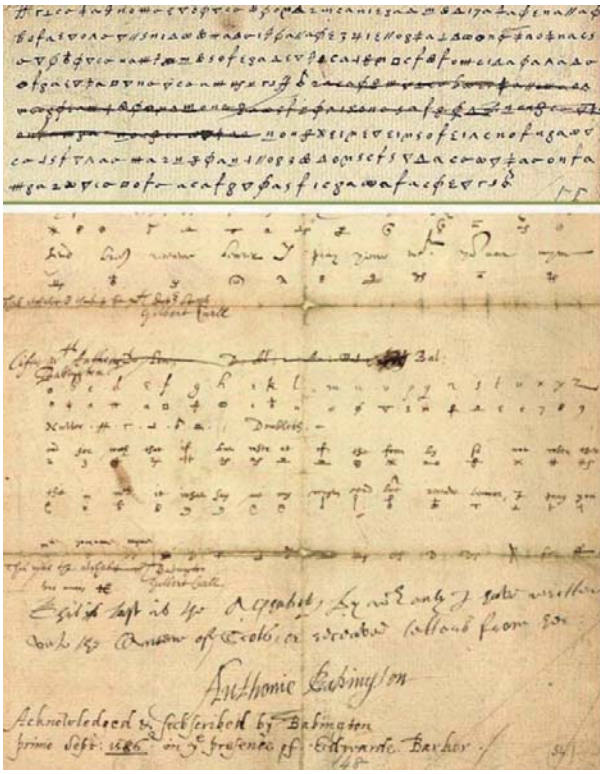

Figura 2. Ejemplo de cifrado con dos alfabetos distintos.

En nuestro caso la letra *A* es codificada como 1 y las demás secuencialmente hasta la letra *Y* como 26. A la letra *Z* le corresponde 0. Con este alfabeto, tanto el texto claro como el texto cifrado son simples secuencias finitas (lista o vectores) de elementos (números) de Ω. Además, este cifrado se define utilizando la aplicación biyectiva:

$$\begin{aligned} f:\ \Omega &\rightarrow \Omega \\ n &\rightarrow f(n) = (n+3) \bmod 27 \end{aligned}$$

donde $(n+3) \bmod 27$ representa el resto de la división entera del número $n+3$ entre 27.

Para obtener de un texto cifrado un texto claro o *texto descifrado* se usa la aplicación inversa de la aplicación de cifrado *f*.

$$f^{-1}: \Omega \rightarrow \Omega$$
$$n \rightarrow f(n) = (n+24) \bmod 27$$

Si bien aquí hemos optado por el conjunto $[0,26]_{\mathbb{N}}$, que es el conjunto de menores residuos no negativos módulo 27 en la aritmética modular, se puede optar por operar con cualquier conjunto completo de residuos módulo 27 a la hora de realizar las operaciones con número enteros módulo 27; por ejemplo, el intervalo de números enteros $[-12,13]_{\mathbb{Z}}$. Recordamos que en $\mathbb{Z}$ se verifica que:

$$(n+24) \bmod 27 = (n-3) \bmod 27.$$

Ahora, el texto claro "CIFRADOSCESARALEATORIOS" se representa (codifica) por la lista de códigos:

[3,9,6,19,1,4,16,20,3,5,20,1,19,1,12,5,1,21,16,19,6,16,20],

y el texto cifrado por:

[6,12,9,22,4,7,19,23,6,8,23,4,22,4,15,8,4,24,19,22,12,19,23],

es decir, "FLIUDGRVFHVDUDÑHDWRULRV".

Se puede definir de forma genérica cualquier cifrado por desplazamiento de salto k empleando la correspondiente aplicación biyectiva de cifrado f_k y su aplicación inversa f_k^{-1} para descifrar. Por ejemplo:

$$f_k: \Omega \rightarrow \Omega \qquad n \rightarrow f_k(n) = (n+k) \bmod |\Omega|$$

y

$$f_k^{-1}: \Omega \rightarrow \Omega \qquad n \rightarrow f_k^{-1}(n) = (n + |\Omega| - k) \bmod |\Omega|$$

Si para unos, los que envían mensajes, lo importante es cifrar aquellos mensajes que generan, para otros, los espías, lo importante es descifran los mensajes ocultos, es decir, romper los cifrados. Hoy es fácil de entender que los cifrados por desplazamientos son fáciles de romper, pero en la época de Cesar no todos sabían leer ni operar como hoy leen y operan los niños. Sorprendentemente los cifrados por desplazamiento siguen siendo usados hoy en día. El propósito actual de este tipo de cifrado no es el de ocultar la información sino de impedir la lectura inmediata de los mensajes. Un cifrado por desplazamiento bastante usado vía internet es el cifrado con desplazamiento $k = 13$, denominado ROT13, lo cual permite cifrar la solución de algún reto junto con su enunciado de manera que quien afronte dicho reto pueda comprobar si su solución es correcta.

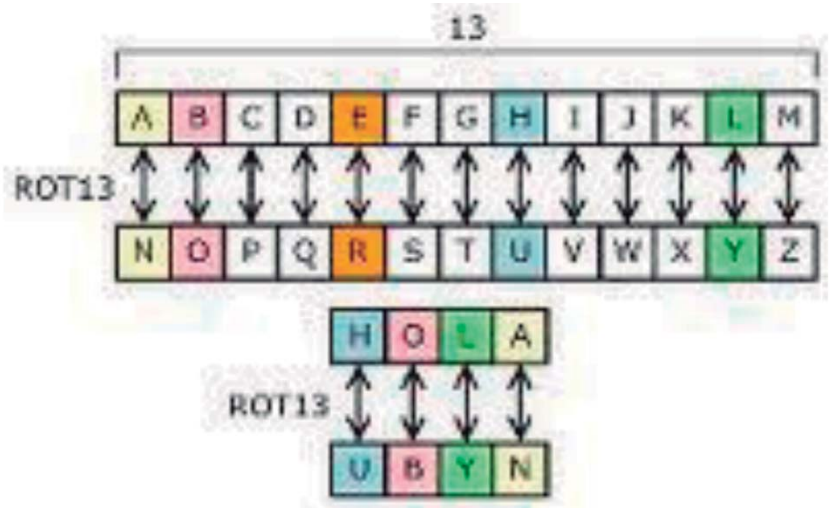

Figura 3. Ejemplo del cifrado ROT13 con alfabeto inglés.

2. CIFRADOS POR SUSTITUCIÓN

Otro cifrado clásico utilizado por el antiguo pueblo judío es el *cifrado Atbash* o *cifrado espejo*. Nosotros lo ejemplificamos con nuestro alfabeto mediante la aplicación:

$$\begin{aligned} f: \Omega &\rightarrow \Omega \\ n &\rightarrow f(n) = (28-n) \bmod 27 \end{aligned}$$

En este caso la función inversa es $f^{-1} = f$, es decir, descifrar es aplicar otra vez el cifrado al texto cifrado.

Dados dos alfabetos Ω y Ω' tales que $|\Omega|=|\Omega|$ y una aplicación biyectiva de Ω a Ω, se puede definir un tipo de cifrado (*cifrados por sustitución)* donde el texto cifrado se construye al sustituir cada elemento del texto claro por su imagen según la aplicación. El cifrado Cesar y el cifrado Atbash son casos particulares de cifrado por sustitución donde $\Omega = \Omega'$. Estos dos cifrados de sustitución son denominados *cifrados de sustitución monoalfabética,* pues realizan la sustitución elemento a elemento en el texto claro. Dado que estamos suponiendo alfabetos numéricos, diremos que son *cifrados escalares de sustitución* pues se basan en la existencia de una aplicación escalar biyectiva de Ω a Ω.

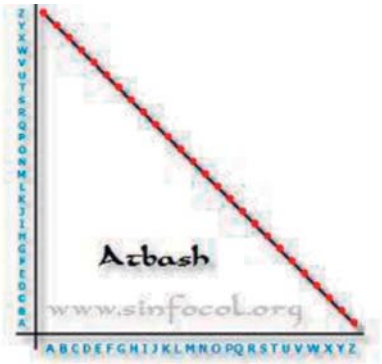

Figura 4. Función cifrado Atbash con alfabeto inglés.

También existen *cifrados vectoriales de sustitución*, que se basan en la existencia de una aplicación vectorial biyectiva de Ω^m a Ω^m. De manera que la sustitución en el texto claro se realiza descomponiendo dicho texto en una secuencia de bloques de m elementos contiguos y sustituyendo cada bloque por su imagen. Estos cifrados son denominados *cifrados de sustitución polialfabéticos*.

El primer cifrado polialfabético conocido data del siglo XV, aunque el paradigma de este tipo de cifrados es conocido como cifrado *Vigenère*. Cifrado del siglo XVI que se consideró irrompible hasta el siglo XIX.

Para definir un cifrado polialfabético sólo se necesita elegir un valor de m y una aplicación biyectiva:

$$\begin{aligned} f: \Omega^m &\rightarrow \Omega^m \\ [n_1,\ldots,n_m] &\rightarrow f([n_1,\ldots,n_m]) \end{aligned}$$

El texto cifrado se produce al trocear el mensaje claro en sucesivos bloques de m elementos y sustituir cada bloque por el bloque imagen dado por la aplicación f.

En el cifrado Vigenère la aplicación biyectiva que se considera es:

$$\begin{aligned} f: \Omega^m &\rightarrow \Omega^m \\ [n_1,\ldots,n_m] &\rightarrow f([n_1,\ldots,n_m]) = [f_{k1}(n_1), \ldots, f_{km}(n_m)] \end{aligned}$$

donde $f_{k_1},\ldots, f_{k_m}$ son m aplicaciones biyectivas que definen m cifrados por desplazamiento de saltos $k_1 \ldots k_m$.

Con este tipo de cifrado resulta que el elemento n situado en el lugar p del texto claro es sustituido por el elemento $f_{k_q}(n)$, , donde $q = p \bmod m$, en el texto cifrado.

Con $m = 4$ y f_8, f_{16}, f_{12}, f_1 tenemos que el anterior texto claro se transforma en el texto cifrado:

[11,25,18,20,9,20,1,21,11,21,5,2,0,17,24,6,9,10,1,20,14,5,5].

Si este último se expresa con letras se obtiene: “KXQSISATKTEBZPWFIJASNEE”.

Para un cifrado Vigenére tan sólo se hace necesario disponer de una palabra clave, en este caso la palabra “HOLA”, pues el valor m y los desplazamientos k_1,k_2,k_3,k_{14} quedan implícitamente definidos por esa palabra o *clave de cifrado*. Al codificar con la palabra “HOLA” se tiene $m = 4$ y $[k_1,k_2,k_3,k_{14}] = [8,16,12,1]$.

Es evidente que la fundamentación matemática del cifrado por sustitución no era lo que los encargados de cifrar y descifrar los mensajes necesitaban en tiempos de Cesar, pero hoy es bueno saberla. A esos cifradores les resultaría muy útil disponer de una herramienta tan elemental como es un disco con el abecedario inscrito en él, como horas en un reloj, de forma que pudieran seguir el desplazamiento de cada letra fácilmente. Mucho mejor sería para los cifradores Vigenère disponer de varios juegos de dos discos concéntricos con los abecedarios en cada uno de ello, el interior para las letras de la clave y el exterior para las letras del texto. Sin duda utilizaban alguna tabla de conversión. Tabla constituida con 27 filas y 27 columnas y donde quedaba reflejado el resultado de la actuación de cada letra de la clave sobre cada letra del mensaje. Hoy simplemente decimos que se trata de una representación alfabética de la tabla de sumar del grupo $G = (\mathbb{Z}/27, +)$, siendo $\mathbb{Z}/27 = \{0,1,2, \ldots 26\}$, y la operación $a + b = c$, siendo $c = (a + b) \bmod 27$.

Figura 5. Ruleta de cifrar.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
01	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
02	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
03	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
04	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
05	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
06	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
07	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
08	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
09	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
10	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
11	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
12	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
13	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
14	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
15	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
16	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
17	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
18	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
19	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
20	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
21	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
22	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
23	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
24	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
25	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
26	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

HTTP://PHYSICAETMATHEMATICA.BLOGSPOT.COM/

Figura 6. Tabla de cifrado Vigenère.

Hoy no necesitamos esas herramientas pues tenemos ordenadores que nos permiten generar un modelo de estos cifrados empleando los fundamentos matemáticos y declarándolos mediante la adecuada programación.

3. CIFRADOS RACIONALES DE SUSTITUCIÓN

En el cifrado de sustitución se tiene que el texto claro, T_c, y el encriptado, T_e , tienen la misma longitud $|T_c|=|T_e|=p$, por ello podemos considerar que ambos son elementos del grupo producto $(G^p,+)$, donde $G^p= G \times G \times ... \times G$ y

$$[a_1,... a_p] + [b_1,... b_p] = [a_1 + b_1,... ,a_p + b_p].$$

Así pues, elegido un determinado elemento $\alpha \in G^p$, un proceso de cifrado por sustitución de $T_e \in G^p$ consiste en operar con ese elemento:

$$T_c + \alpha = T_e.$$

El descifrado de $T_e \in G^p$, es operar con el opuesto del elemento de α; $T_e - \alpha = T_c$.

En los ejemplos de cifrados Cesar y Vigenère se eligieron los elementos:

$$\alpha_C = [3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3]$$

$$\alpha_V = [8,16,12,1,8,16,12,1,8,16,12,1,8,16,12,1,8,16,12,1,8,16,12]$$

El valor de p no es fijo, puesto que es la longitud del texto claro y del cifrado. Si pensamos en un texto de longitud no finita, sin duda tanto el texto claro como el cifrado son sucesiones de elementos de G y, por lo tanto, para cifrar se requerirá determinar una sucesión con la que operar.

La sucesión del cifrado Cesar del ejemplo es $\alpha_C = \{a_n = 3\}_{n\in\mathbb{N}}$, una sucesión constante. Las de los cifrados Vigenére son sucesiones alternas tomando un número finito de valores distintos. En el ejemplo es:

$$\alpha_V = \{a_{4n} = 8 \ , a_{4n+1} = 16 \ , \ a_{4n+2} = 12 \ , a_{4n+3} = 1,\}_{n\in\mathbb{N}}.$$

Tanto la sucesión α_c como la sucesión α_v están asociadas a dos números racionales cuya expresión "decimal periódica pura", expresada en un sistema de numeración base 27, es $r_C = 0.3\ 3\ 3\ 3\ 3\ 3_{27}...$ y $r_V = 0.8\ 16\ 12\ 1\ 8\ 16\ 12\ 1\ 8_{27}...$, respectivamente.

Las expresiones en forma de fracción en base 27 son $r_c = \dfrac{3_{27}}{26_{27}}$ y $r_v = \dfrac{8\ 16\ 12\ 1_{27}}{26\ 26\ 26\ 26_{27}}$, mientras que en base 10 son:

$$r_c = \frac{3}{26} \quad \text{y} \quad r_v = \frac{8\ 27^3+16\ 27^2+12\ 27+1}{26\ 27^3+26\ 27^2+26\ 27+26} = \frac{169453}{561440}$$

En estos dos casos, en lugar de disponer de la clave inicial (sucesión), se puede utilizar un par de números, el numerador y el denominador asociados a cada cifrado, es decir, el correspondiente número racional.

Sin duda son cifrados de sustitución aquellos que se corresponden con cualquier número racional, pues una vez establecida su expresión decimal en base 27, si tiene una expresión periódica mixta el tipo de cifrado sería como un cifrado Vigenère que se inicia en el carácter en la posición siguiente a la longitud de la parte no periódica del número. Estos cifrados son una combinación de cifrados de sustitución y los denominamos *cifrados racionales de sustitución.*

La utilización manual de una expresión "decimal" en base 27 puede ser un engorro, aunque un pequeño programa de ordenador nos facilite operar en base 27. Por ello, a continuación presentamos una pequeña variante de cifrados racionales de sustitución. Se utiliza simplemente la expresión decimal en base 10 del número racional, por ejemplo, de r = 121/372, su expresión decimal r = 0.32526881720430107526881...

De la parte decimal periódica, 881720430107526, se agrupan de dos en dos los decimales hasta que se pueda o hasta el final, es decir, 88, 17, 20, 43, 01, 07, 52, 6 , y transforman modúlo 27 para formar la clave numérica [7,17,20,16,1,7,25,6], o lo que es lo mismo "GPSOAGXF". Se procede de forma análoga con la parte no periódica 3252 y obtenemos [14, 15] . Con estas dos partes construimos la sucesión α_r que utilizamos.

$$\alpha_r=\{14,25,7,17,20,16,1,7,25,6,7,17,20,16,1,7,25,6,7,17,20,16,1...\},$$

Al operar el texto claro con el vector compuesto por los veintitrés primeros términos:

$$14,25,7,17,20,16,1,7,25,6,7,17,20,16,1,7,25,6,7,17,20,16,1$$

se obtiene el texto cifrado:

[17,7,13,9,21,20,17,0,1,11,0,18,12,17,13,12,26,0,23,9,26,5,21],

o lo que es lo mismo: “PGMITSPZAKAQLPMLYZVIYET”.

Ya hemos indicado que el cifrado Vigenère no fue roto hasta el siglo XIX. Lo rompió por primera vez un militar prusiano F. Kasiski. Su método consiste en determinar la longitud de la clave y se basa en la búsqueda de agrupaciones alfabéticas repetidas en el texto cifrado.

Estas repeticiones indican con alta probabilidad que dichas agrupaciones no sólo se obtienen de grupos idénticos del texto claro, en los cuales la clave coincide en la misma posición en esas ocurrencias.

Como la distancia entre palabras repetidas es múltiplo de la longitud de la clave, es cuestión de buscar las distancias de repetición de diferentes palabras que se puedan repetir y hallar el máximo común divisor para poder establecer un múltiplo próximo a la longitud de la clave. La longitud de la clave será este número o algún factor primo del mismo.

Conocida la longitud de la clave, se divide el texto en bloques del tamaño de la clave y se aplica el método estadístico tradicional.

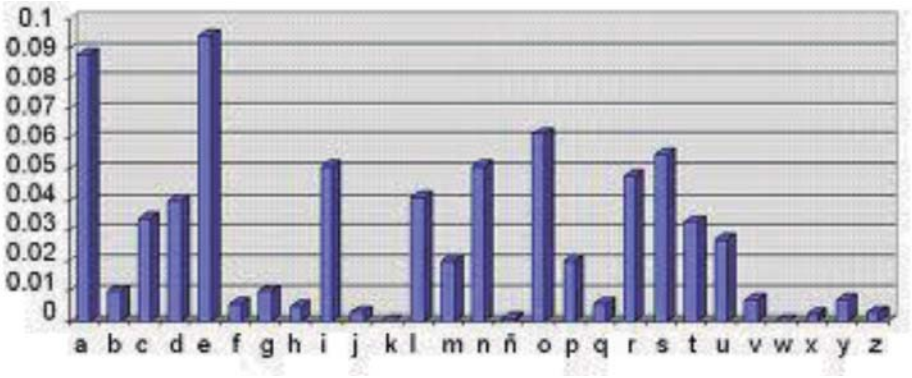

Figura 7. Probabilidad de aparición de cada letra en español.

4. CIFRADOS IRRACIONALES DE SUSTITUCIÓN

Es claro que el método Kasiski permite romper cualquier texto cifrado obtenido por cifrado racional de sustitución para claves de tamaños relativamente pequeños en comparación con el tamaño del texto claro. Ahora bien, si el tamaño de la clave y el del texto claro son iguales, entonces el método Kasiski no es operativo y el cifrado es “seguro”.

Figura 8. Máquina de cifrado Enigma.

A continuación presentamos la forma de obtener claves de cualquier tamaño deseado que permita la encriptación segura de los textos claros. En esencia es una generalización de los métodos racionales de cifrados por sustitución aludidos en el apartado anterior. En lugar de elegir un número racional en modo de fracción y generar su expresión decimal, se elige un número irracional y se utiliza directamente su expresión decimal.

El motivo de la elección de un número irracional es debido a que su expresión decimal no es periódica, y puede elegirse un número racional con expresión decimal finita, tan próximo al irracional como se desee, es decir, se pueden elegir tantos decimales como sean necesarios en el momento de cifra del mensaje. Para nuestro ejemplo podemos emplear el número irracional $r = \sqrt{2}$ y elegimos una aproximación decimal finita con el doble de decimales que la longitud del texto claro, en este caso como el texto claro tiene una longitud de 23 caracteres consideramos los primeros 50 decimales:

1.41421356237309504880168872420969807856967187537 69.

Se procede con la parte decimal de la misma forma que se hizo con la parte periódica del número racional del apartado 3, es decir, agrupando de dos en dos los decimales:

[41,42,13,56,23,73,09,50,48,80,16,88,72,42,09,69,80,78,56,96,71,87,53]

y se construye, sustituyendo cada número por su resto módulo 27, el vector necesario para obtener el texto cifrado.

α=[14,15,13,2,23,19,9,23,21,26,16,7,18,15,9,15,26,24,2,15,17,6,26]

Del texto claro: [3,9,6,19,1,4,16,20,3,5,20,1,19,1,12,5,1,21,16,19,6,16,20], obtenemos el texto cifrado: [17,24,19,21,23,25,16,24,4,9,8,10,16,21,20,0,18,18,7,26,22,19], o lo que es lo mismo: "PWRTWVXOWDIHJOTSZQQGYUR".

Con este tipo de cifrado hemos utilizado unas claves (vectores de cifrado) que poseen la misma longitud que el texto que se desea cifrar. Que las longitudes de la clave y del texto claro sean iguales es una característica muy importante, puesto que es una de las condiciones que propuso Gilbert Vernon en 1917 cuando patentó su cifrado. Del cifrado Vernon podemos asegurar que es irrompible, pues Claude Shannon lo demostró entre los años 1940 y 1945.

Si algún espía tuviera acceso al vector de cifrado de un cifrado irracional, podría pensar que las componentes de dicho vector constituyen una secuencia aleatoria de códigos, cosa que no lo es. La aleatoriedad de la clave es otra de las características de las claves del irrompible cifrado Vernon. Nosotros optaremos por conseguir un cifrado irracional que posea esta otra característica. En esencia, nos preocuparemos de la forma de obtener un número irracional adecuado y no un número "sencillo" como raíz de dos.

El disponer de un número irracional con tantos decimales como se deseen no era una cuestión sencilla en tiempos ni de Vernon ni de Shanon, pero hoy disponemos de ordenadores que pueden facilitar tales vectores. Por ejemplo, la instrucción del Maple: *evalf* [50](*sqrt*(2)) da 1.41421356237309504880168872420969807856967187537 69.

Si se sustituye el número 50 por el número 500000 no hay ningún problema. Espero que el lector no me crea y lo pruebe, pues se sorprenderá de la velocidad del cálculo y de las pantallas que rellena la respuesta.

5. CIFRADOS CESAR ALEATORIOS

Como ya hemos indicado, nos preocupa la forma de elegir el número irracional que emplearemos en el cifrado irracional. No está mal emplear el número irracional anterior, pero sería mejor cambiar de número irracional, y por tanto de clave. Por otro lado, no parece lógico cambiar constantemente de número pues el receptor debe saber que se ha cambiado, y hay que comunicárselo de alguna forma para que pueda descifrar los mensajes que le lleguen. ¿Estarán los espías esperando algún fallo de seguridad? ¿Dónde es más fácil que se produzca ese fallo? Sin duda en la comunicación del cambio de número irracional.

Es el momento de proponer un reto al lector. Considere el cifrado irracional correspondiente al número raíz de dos y obtenga tantos decimales de dicho número como sean necesarios. Construya un vector de cifrado de la siguiente forma: Una vez construido un vector como se indica en el apartado 4, se eliminan aquellas componentes de ese vec-

tor que ocupan una posición correspondiente a un número primo y se dejan las restantes. ¿Constituyen las coordenadas de este nuevo vector una secuencia suficientemente pseudoaleatoria? ¿Se le ocurre otro mecanismo para eliminar componentes del vector inicial y obtener un nuevo vector de cifrado? Supongo que sí. Además, pienso que puede idear una secuenciación adecuada para cambiar el método de eliminación de componentes.

Al utilizar alguna forma de eliminar componentes del vector inicial que no presenta el cifrado irracional se nos presenta otra de las características esenciales del cifrado Vernon. Esa característica es que la clave sólo se utiliza una vez para cifrar un texto. Eso es claro, pues el siguiente cifrado empleará otro vector clave aunque haya partido del mismo vector inicial.

El problema esencial del cifrado irracional es elegir el número irracional y formas de construir números irracionales hay tantas como números irracionales hay. Esta afirmación deberíamos demostrarla, pero no aquí.

Éste es el momento de crear un vector de las características del vector de cifrado de un encriptado irracional. Para ello emplearemos las características caóticas que el método de iteraciones sucesivas puede presentar con algunas funciones. Es decir, utilizaremos un proceso de iteración que produzca caos para generar secuencias pseudoaleatorias de códigos, tal y como me indicó el profesor Antoranz.

Elegida una función, por ejemplo:

$$\begin{aligned} f : \mathbb{R} &\to \mathbb{R} \\ x &\to f(x) = kx\,(1\text{-}x) \end{aligned}$$

donde se elige un número $k \in \mathbb{R}$, el proceso de iteraciones sucesivas consiste en construir una sucesión de número reales $\{x_n\}_{n\in\mathbb{N}}$ donde x_0 es un valor inicial elegido y el resto se define de la forma $x_{(n+1)} = f(x_n)$.

Ahora bien, si restringimos el dominio de la función al intervalo [0,1] tenemos que:

$$\begin{aligned} f\text{:}\ [0,1] &\to [0,1] \\ x &\to f(x) = kx\,(1\text{-}x)) \end{aligned}$$

con $k \in (0,4)$. Así pues, la sucesión de número reales $\{x_n\}_{n\in\mathbb{N}}$ está compuesta por números entre cero y uno.

Nos interesa crear una sucesión de códigos $\{c_n\}_{n\in\mathbb{N}}$ para poder extraer un vector de cifrado de la dimensión que deseemos. Definimos como:

$$c_n = [|\Omega| f(x_n)] \text{mod } |\Omega|$$

donde $[t]$ representa la parte entera del número t y dado el alfabeto Ω. En nuestro caso, $c_n = [27f(x_n)]\text{mod } 27$.

Hemos elegido esta función por el comportamiento caótico de los valores obtenidos por iteración sucesiva cuando se toma cualquier valor de k tal que $3.7 \leq k < 4$. ¿Qué queremos decir con esto de "comportamiento caótico"? Pues que la sucesión de número reales $\{x_n\}_{n\in\mathbb{N}}$ no converge, y no sólo eso, sino que "toma valores aparentemente de forma aleatoria". También ocurre que al construir las sucesiones $\{x^1_n\}_{n\in\mathbb{N}}$ y $\{x^2_n\}_{n\in\mathbb{N}}$ correspondientes a dos valores del parámetro, k_1 y k_2 muy cercanos, éstas difieren totalmente. Conocer el valor del parámetro de una sucesión no aporta información alguna sobre el valor de otro parámetro. Algo análogo ocurre con la variación del valor inicial elegido. En definitiva, si queremos construir correctamente una parte de la sucesión necesitamos conocer el valor del parámetro y algún término de la sucesión.

Desde un punto de vista gráfico, cada uno de los puntos $(x_n, f(x_n))$ está situado en la parábola $y = kx - kx^2$, pero para esos valores del parámetro y a la hora de situarlos secuencialmente sobre la gráfica, parece que saltan de un lado a otro sobre dicha gráfica. Parece que van a rellenarse todos los puntos de la parábola, pero no se tiene ni idea en el orden de colocación. Le aconsejamos que experimente con unas 300 iteraciones.

El método de cifrado Cesar aleatorio que proponemos se basa en emplear un vector de cifrado obtenido de la forma similar a la indicada anteriormente partiendo de un valor del parámetro k y un valor inicial x_0.

Como ejemplo hemos utilizado los valores $k = 3.945$ y $x_0 = 1/9$, obteniendo el vector de cifrado eligiendo valores después de las 100 primeras iteraciones iniciales:

$$\alpha = [10,25,6,19,21,16,25,6,20,19,20,18,22,14,26,1,6,19,21,17,24,9,24].$$

Al utilizar el cifrado correspondiente al vector resulta que nuestro texto claro inicial:

[3,9,6,19,1,4,16,20,3,5,20,1,19,1,12,5,1,21,16,19,6,16,20],

se transforma en el texto cifrado: [13,7,12,11,22,20,14,26,23,24,13,19,14,15,11,6,7,13,10,9,6]. Es decir, en "MGLKUSNYVWMRNÑKFGMJIFXP".

Obsérvese que hemos descrito un cifrado que tiene la longitud de la clave del tamaño del texto claro, que la secuencia de códigos es "aleatoria" y que, si así lo queremos, cada clave sólo la utilizamos una vez. Una vez utilizada la clave basta conservar el valor obtenido de la última iteración, valor que servirá como valor inicial en el siguiente cifrado.

¿Es este tipo de cifrado de Cesar aleatorio un cifrado del tipo Vernon?, es decir, ¿existe alguna biyección de un cifrado Cesar aleatorio a un cifrado Vernon? La respuesta corresponde a otro trabajo.

6. CÓDIGO MAPLE DEL LABORATORIO DE CIFRADOS

Los párrafos que empiezan con el símbolo # son simples comentarios explicativos para el programador, mientras que los que comienzan con > son líneas de comandos. Estas últimas son las estrictamente necesarias y las remarcamos en negrita.

```
# Laboratorio de Cifrados Cesar, Vigenére, sustitución irracional y Cesar aleatorio.
#
> restart; with(StringTools);
# Bloque 1: Alfabeto considerado
# Alfabeto que se considera:
> Abecedario := "ABCDEFGHIJKLMNÑOPQRSTUVWXYZ";
> AlfabetoInicial := convert(Abecedario, list): jc := nops(AlfabetoInicial):
# Función Resto
> reduce := x -> modp(x, jc):
# Codificación letra: Asigna un número (código) a una letra
> Codificar:=x->if x=209 then 15 elif x<79 then x-64: else reduce(x-63);end if:
# Decodificació numero: Asigna una letra a un código
> Decodificar:=x->if x=0 then 90 elif x=15 then 209 elif x<15 then x+64: else x+63:
 end if:
# Codificación de texto: Transformaccio una cadena de texto en una lista de números (códigos)
> codificatexto := proc (cadena) local listacadena, textoclaro, i, letra;
 listacadena := convert(cadena, list); textoclaro := [ ];
 for i to nops(listacadena) do
 letra := op(i, listacadena); textoclaro := [op(textoclaro), Codificar(Ord(letra))] :
 end do;
 return (textoclaro ):
 end proc;
# Decodificación de lista de códigos: Transforma una lista de números en una lista de caracteres (tipo=1) o una cadena (tipo=0)
> decodificanumeros := proc (listanum, tipo) local mensaje, texto, i, numero;
 mensaje := ""; texto := [ ];
 for i to nops(listanum) do
 numero := op(i, listanum); texto := [op(texto), Char(Decodificar(numero))];
 mensaje := cat(mensaje, Char(Decodificar(numero)));
 end do;
```

```
 if tipo = 1 then return texto else return mensaje end if;
 end proc;
# Bloque 2: Entrada de Datos
# Cifrar (CD=1), Descifrar (CD=0) y Cifrar+Descifrar (CD=2)
> CD := 2;
# Texto que se desea cifrar o descifrar:
> textoinicial := “CIFRADOSCESARALEATORIOS”;
# Tipo de cifrado Cesar Clásico (TC=1), Vigenère (TC=2), Irracional (TC=3), Cesar Aleatotio (TC Resto de valores)
> TC := 3;
# Elementos necesarios para Sustitución Vigenère
# Palabra clave para cifrar:
> clave := “HOLA”;
# Elementos necesarios para Sustitución Irracional
# Número irracional
> numirrac := sqrt(2);
# Elementos necesarios para Sustitución Cesar Aleatorio
# factor de caos 3.7<k<4:
> k:=3.945:
# Valor inicial
> ValorInicial := 1/9;
# Ver el alfabeto utilizado Si (Alf=1) No (Alf Resto de valores)
> Alf := 0;
# Bloque 3: Modulo de control de salto según cifrado
# Tipo de cifrado
> if CD = 0 then textocifrado := codificatexto(textoinicial);
 ltc := nops(textocifrado):
 else textoclaro := codificatexto(textoinicial); ltc := nops(textoclaro)
 end if;
# Cálculos para Vigenère
# Creación de un vector de códigos:
> ClaveClaro := codificatexto(clave);
# Longitud de la clave:
> lc:=nops(ClaveClaro):
# Cálculos para Sustitución Irracional
# Decimales del número
> cadenadecimales := convert(evalf[2*ltc+11](numirrac), string);
 lista := convert( cadenadecimales, list);
> textoraiz := [ ];
 for i from 3 to nops(lista) do
 num := op(i, lista);
 textoraiz := [op(textoraiz), Ord(num)-48];
 end do; textoraiz;
# Creación de un vector de códigos:
> ClaveIrracional := [ ];
 for i from 3 by 2 to nops(lista) do
 num1 := op(i, lista); num := op(i+1, lista);
 ClaveIrracional := [op(ClaveIrracional), modp(10*(Ord(num1)-48)+Ord(num)-48, 27)];
 end do;
```

```
 ClaveIrracional: nops(ClaveIrracional):
# Cálculos para Cesar Aleatorio
# Función generadora de numeros
> f := x-> k*x*(1-x);
# Inicio de valores
> control:=ValorInicial: for i from 1 by 1 to 100 do control:=f(control): end do:
# Creación de un vector de códigos
> ClaveCAleatorio := [ ];
 for i to nops(textoclaro) do
 ClaveCAleatorio := [op(ClaveCAleatorio), modp(floor(control*jc), 27)];
 control := f(control);
 end do;
 ClaveCAleatorio; nops(ClaveCAleatorio);
# Control del desplazamiento de cada elemento del texto
> salto := proc (tc, i) local indice;
 if tc = 1 then return 3 elif tc = 2 then indice := if (modp(i, lc) = 0, lc, modp(i, lc));
 return op(indice, ClaveClaro); elif tc = 3 then return op(i, ClaveIrracional);
 else return op(i, ClaveCAleatorio);
 end if; end proc;
# Módulo de Cifrado y Descifrado
# Procedimiento de Cifrado o Encriptado
> cifrar := proc (texto, tc) local VCifrador, TCifrado, saltocesar, i;
 VCifrador := [ ]; TCifrado := [ ];
 for i while nops(VCifrador) < nops(texto) do
 saltocesar := salto(tc, i); VCifrador := [op(VCifrador), saltocesar];
 TCifrado := [op(TCifrado), reduce(saltocesar+op(i, texto))];
 end do;
 return [VCifrador, TCifrado] :
 end proc;
# Procedimiento de Descifrado o Desencriptado
> descifrar := proc (texto, tc) local VCifrador, TDescifrado, saltocesar, i;
 VCifrador := [ ]; TDescifrado := [ ];
 for i while nops(VCifrador) < nops(texto) do
 saltocesar := salto(tc, i); VCifrador := [op(VCifrador), saltocesar];
 TDescifrado := [op(TDescifrado), reduce(-saltocesar+op(i, texto))]
 end do;
 return [VCifrador, TDescifrado];
 end proc;
# Salida de datos
> if TC = 1 then print(“”); print(“Cifrado Cesar Clásico”); print(“”);
 elif TC = 2 then print(“”); print(“Cifrado Vigenére”); print(“”);
 elif TC = 3 then print(“”); print(“Cifrado Sustitucón Irracional”); print(“”);
 else print(“”); print(“Cifrado Cesar Aleatorio”); print(“”);
 end if;
> if Alf = 1 then Ncaracteres = jc; AlfabetoLetras = AlfabetoInicial;
 AlfabetoNumerico := codificatexto(Abecedario); print(“”);
 print(“Alfabeto considerado :”, AlfabetoInicial, “ de ”, jc, “ letras”);
 print(“Códigos numericos : ”, AlfabetoNumerico);
```

```
 end if;
> if CD = 0 then salida := descifrar(textocifrado, TC); vectorcifrar = op(1, salida);
 TextoDescifrado := op(2, salida);
 TextoLetras := decodificanumeros(TextoDescifrado, 0);
 print("Vector de Cifrado= ", vectorcifrar); print("");
 print("Texto Descifrado= ", TextoDescifrado);
 print("Cadena Descifrada= ", TextoLetras);
 end if;
> if CD = 1 then salida := cifrar(textoclaro, TC); vectorcifrar := op(1, salida);
 TextoCifrado := op(2, salida);
 TextoLetras := decodificanumeros(TextoCifrado, 0);
 print(" Vector de Cifrado= ", vectorcifrar);
 print(""); print("Texto Cifrado= ", TextoCifrado);
 print("Cadena Cifrada= ", TextoLetras);
 end if;
> if CD = 2 then salida := cifrar(textoclaro, TC); vectorcifrar := op(1, salida);
 TextoCifrado := op(2, salida);
 TextoLetras := decodificanumeros(TextoCifrado, 0);
print("Vector de Cifrado= ", vectorcifrar); print("");
print("Texto Cifrado= ", TextoCifrado); print("Cadena Cifrada= ", TextoLetras);
salida := descifrar(TextoCifrado, TC); vectorcifrar = op(1, salida);
TextoDescifrado := op(2, salida);
TextoLetras := decodificanumeros(TextoDescifrado, 0);
print(""); print("Vector de Cifrado= ", vectorcifrar); print("");
print("Texto Descifrado= ", TextoDescifrado);
print("Cadena Descifrada= ", TextoLetras);
end if;
```

BIBLIOGRAFÍA

[1] *Cryptography. Theory and Practice*. Douglas R. Stinson. CRC Press, Florida, USA (1995). ISBN: 978-08493-8521-6.

[2] *Estructuras fractales y sus aplicaciones*, Miguel de Guzmán *et al.* Ed. Labor, Barcelona, España (1993). ISBN: 84-335-5152-3.

[3] *Cryptology*. Albrecht Beutelspacher. Mathematical Association of America, Washington, USA (1994). ISBN: 978-08838-5504-6.

[4] *Técnicas Criptográficas de protección de datos*. Amparo Fúster Sabater *et al.* Ed. Rama, Madrid, España (1997). ISBN: 978-84789-7594-5.

[5] www.wikipedia.org.

Miguel Delgado Pineda
miguel@mat.uned.es
Dpto. de Matemáticas Fundamentales