CRIPTOGRAFÍA EN EL AULA DE MATEMÁTICA

Betina Russo
betinaru@yahoo.com.ar

Resumen
En este artículo se explican algunas técnicas sencillas de encriptación y se propone su enseñanza en el aula como ejemplos de aplicación en las clases de computación. Se incluye también un programa desarrollado bajo Excel que permite cifrar un mensaje con una de las técnicas descriptas.

INTRODUCCIÓN

La Criptografía estudia las maneras de cifrar mensajes. Esto se hace desde tiempos muy antiguos. Ya en el siglo I antes de Cristo Julio César se comunicaba con sus generales en batalla con mensajes cifrados. Utilizaba un sistema que consistía en desplazar cada letra cuatro posiciones hacia delante en el alfabeto. Por eso, algunos libros lo mencionan como clave Cesárea.

También los hebreos cifraban textos, según lo menciona la Biblia, mediante el uso del alfabeto invertido. Reemplazaban la primera letra del alfabeto por la última, la segunda, por la penúltima, etc. Este sistema se denomina Atbash.

En *El escarabajo de oro*, Edgar Allan Poe también hizo uso de la criptografía. En el relato aparece un mensaje en clave, que logra ser descifrado, y una minuciosa explicación de las reglas usadas para lograrlo.

A continuación se describirán algunas técnicas sencillas de encriptación que han sido utilizadas históricamente. Como se verá, pueden ser explicadas y puestas en práctica en el aula, aprovechando la utilización de recursos informáticos elementales en la clase de matemática, que sin duda despertarán el interés de los alumnos.

ENCRIPTACIÓN POR DESPLAZAMIENTO

Consiste en reemplazar cada letra del mensaje original por la que resulta de desplazarnos en el alfabeto n caracteres hacia delante. Por ejemplo, supongamos que queremos cifrar la palabra *ORO* con un desplazamiento n=2. Entonces, la *O* se reemplaza por la *Q*, ya que la Q está dos lugares más

adelante en el alfabeto. De la misma manera, la *R* se reemplaza por la *T*. De modo que la palabra *ORO* queda cifrada como *QTQ*.

Si el alfabeto tiene 27 letras, tenemos 26 opciones de encriptación, dado que desplazarse 27 posiciones volvería a dar la misma letra.

ENCRIPTACIÓN POR TABLA

A cada letra del alfabeto se le asigna un símbolo o simplemente otra letra del mismo alfabeto.

Si optamos por usar letras del mismo alfabeto para hacer los reemplazos, vamos a tener tantas posibilidades como formas de ordenar el alfabeto haya. Pensemos en el alfabeto en forma de lista, un símbolo debajo del otro, y al lado, el mismo alfabeto en un orden cualquiera (distinto al original, por supuesto). De esta manera cada símbolo queda emparejado con algún otro. Dependiendo de cómo ordenemos el alfabeto, vamos a obtener una manera diferente de encriptar. Hay que calcular de cuántas maneras distintas es posible ordenar las letras del alfabeto y sabremos ese número. Si el alfabeto tiene 27 letras, esa cantidad se conoce como *permutaciones de 27* y su valor es el factorial de 27 que se obtiene multiplicando 27 x 26 x 25 x … x 2 x 1. Es un número muy muy grande.

Por eso, este método es seguro, dado que es imposible descifrar un mensaje si no se conoce el ordenamiento usado. Probar con todas las posibilidades llevaría demasiado tiempo incluso para una computadora.

Sin embargo es posible descifrar un mensaje sin tener que probar todas las variantes. Esto se logra analizando el mensaje.

Primero hay que saber en qué idioma esta escrito el mensaje original. Luego, teniendo en cuenta la frecuencia de aparición de cada letra, los monosílabos, los prefijos y sufijos, las letras dobles, etc, se van obteniendo las representaciones de algunas letras. También ayuda conocer el tema del mensaje tratando de ubicar palabras que seguramente tienen que aparecer.

En *El escarabajo de oro*, el autor hace uso de todas esas reglas para desencriptar el mensaje. El personaje que en el relato explica como logró descifrarlo dice:

> *"…mi primera misión era averiguar las letras predominantes, así como las que se encontraban con menor frecuencia. Las conté todas y después formé la siguiente tabla […] La letra que se encuentra con mayor frecuencia en un texto en inglés es la e […] Como el signo predominante es el 8, empezaremos por ajustarlo a la e del alfabeto natural. Para comprobar esta suposición, observemos si el 8 aparece a menudo por pares, pues la e se dobla con gran frecuencia en inglés […] Ahora, de todas la palabras de la lengua, the es la más usual; por tanto, debemos ver si no está repetida la combinación de tres signos, siendo el último de ellos el 8…"*

Un ejemplo de este método es el Atbash, que pone en correspondencia al alfabeto original con el mismo, pero invertido.

A la letra A, que es la primera, se le asigna la Z que es la última. A la B, que es la segunda, le asigna la Y, que es la penúltima. Y así sucesivamente.

La tabla quedaría entonces de la siguiente manera:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z
Z	Y	X	W	V	U	T	S	R	Q	P	O	Ñ	N	M	L	K	J	I	H	G	F	E	D	C	B	A

Por ejemplo, para cifrar la palabra *POZO* se reemplaza cada letra por su correspondiente en la tabla. Quedando *KLAL*.

ENCRIPTACIÓN CON PALABRA CLAVE EN COMPUTADORAS

Según este método, el mensaje cifrado se obtiene a partir de una palabra clave. Se arma una lista con todas las letras del mensaje original y se pone en paralelo con otra lista formada por las letras de la palabra clave repetida tantas veces como sea necesario. Por ejemplo, si la palabra clave es *SOL* y el mensaje que se quiere enviar es *TE HARÁS RICO*, la tabla quedaría:

T	S
E	O
H	L
A	S
R	O
A	L
S	S
R	O
I	L
C	S
O	O

Ambas columnas deben ser reemplazadas luego por tiras de bits. Cada letra será reemplazada por su identificación en bits. Luego se aplica entre ambas columnas el operador *or excluyente* conocido como *XOR.*

La columna resultante de la operación, que está en bits, se traduce a símbolos siendo ése el mensaje cifrado. El receptor, conociendo la palabra clave, vuelve a armar la tabla con la operación *XOR* recuperando de esta manera el mensaje original.

Para el ejemplo vamos a tomar un alfabeto en el que cada letra se representa con cinco bits. Codificaremos la palabra *RICO* con la palabra clave *SOL*.

El alfabeto con la representación en bits que vamos a usar es:

A	0	0	0	0	0		P	1	0	0	0	0
B	0	0	0	0	1		Q	1	0	0	0	1
C	0	0	0	1	0		R	1	0	0	1	0
D	0	0	0	1	1		S	1	0	0	1	1
E	0	0	1	0	0		T	1	0	1	0	0
F	0	0	1	0	1		U	1	0	1	0	1
G	0	0	1	1	0		V	1	0	1	1	0
H	0	0	1	1	1		W	1	0	1	1	1
I	0	1	0	0	0		X	1	1	0	0	0
J	0	1	0	0	1		Y	1	1	0	0	1
K	0	1	0	1	0		Z	1	1	0	1	0
L	0	1	0	1	1		*	1	1	0	1	1
M	0	1	1	0	0		#	1	1	1	0	0
N	0	1	1	0	1		&	1	1	1	0	1
Ñ	0	1	1	1	0		%	1	1	1	1	0
O	0	1	1	1	1		$	1	1	1	1	1

Tenemos que armar la tabla con los bits de la palabra *RICO* y de la palabra clave *SOL* para aplicar el operador *XOR*.

Según la tabla, la R en bits es 10010. La *I* es 01000. Hacemos lo mismo con todas las letras. Luego completamos operando con *XOR* y queda la siguiente tabla:

		XOR		
R	1	0	1	S
	0	0	0	
	0	0	0	
	1	0	1	
	0	1	1	
I	0	0	0	O
	1	0	1	
	0	1	1	
	0	1	1	
	0	1	1	

C	0	0	0	L
	0	1	1	
	0	0	0	
	1	0	1	
	0	1	1	
O	0	1	1	S
	1	1	0	
	1	1	0	
	1	0	1	
	1	0	1	

Veamos los primeros cinco bits de la columna del medio, son 00001.

Observando la tabla del alfabeto vemos que esta cadena de bits representa a la letra B. La segunda cadena de cinco bits que quedo es 00111 que representa a la letra H. Las ultimas dos cadenas representan a la letra J y al símbolo #.

La palabra *RICO* queda entonces encriptada como *BHJ#***.**

Quien reciba el mensaje deberá armar su tabla con la palabra BHJ# para obtener la original. Obtendrá la siguiente tabla:

		XOR		
B	0	1	1	S
	0	0	0	
	0	0	0	
	0	1	1	
	1	0	1	
H	0	0	0	O
	0	1	1	
	1	0	1	
	1	0	1	
	1	0	1	

J	0	0	0	L
	1	0	1	
	0	0	0	
	0	1	1	
	1	0	1	
#	1	0	1	S
	1	1	0	
	1	1	0	
	0	1	1	
	0	1	1	

Los primeros cinco bits de la columna del medio son 10010 que representan a la letra *R*. Luego, la cadena 01000 representa la letra *I*. Las dos últimas cadenas representan las letras *C* y *O*. Quedó así la palabra *RICO*, que es justamente la original que estábamos buscando.

ENCRIPTANDO CON EXCEL

La siguiente propuesta se basa en la programación de uno de los métodos de encriptación descriptos. En ella vamos a usar Excel para hacer un programa que nos permita encriptar y desencriptar mensajes rápidamente. Hemos incorporado en este artículo la descripción del programa para permitir a los lectores que lleven a cabo la propuesta.

El objetivo de Excel no es la programación, pero de todos modos, incluye módulos que nos permiten programar en Visual Basic, haciendo al programa mucho más poderoso de lo que ya es. Para saber cómo acceder a estos módulos consultar el libro de Claudio Sánchez (III).

El método que vamos a programar es el Atbash, que reemplaza la primera letra del alfabeto por la última, la segunda por la penúltima, etc.

IDEA GENERAL DEL PROGRAMA

La idea es tomar cada letra del mensaje y según la posición que ocupe en el alfabeto, buscar la letra que debe reemplazarla.

Vamos armando el mensaje cifrado concatenando las letras que van apareciendo en los reemplazos.

¿Cómo sabe el programa que posición ocupa una letra en el alfabeto?

Una forma fácil de guardar las posiciones de las letras de un alfabeto es cargarlas en un vector. De esta manera, el subíndice, nos indicará la posición.

¿Cómo calculamos la posición de la nueva letra?

El alfabeto castellano tiene 27 letras. La letra que ocupa la posición **1** será reemplazada por la que ocupa la posición **27**. La que ocupa la posición **2**, por la que ocupa la posición **26**. La de posición **3** por la de posición **25**, etc.

Armamos una tabla con las posiciones.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

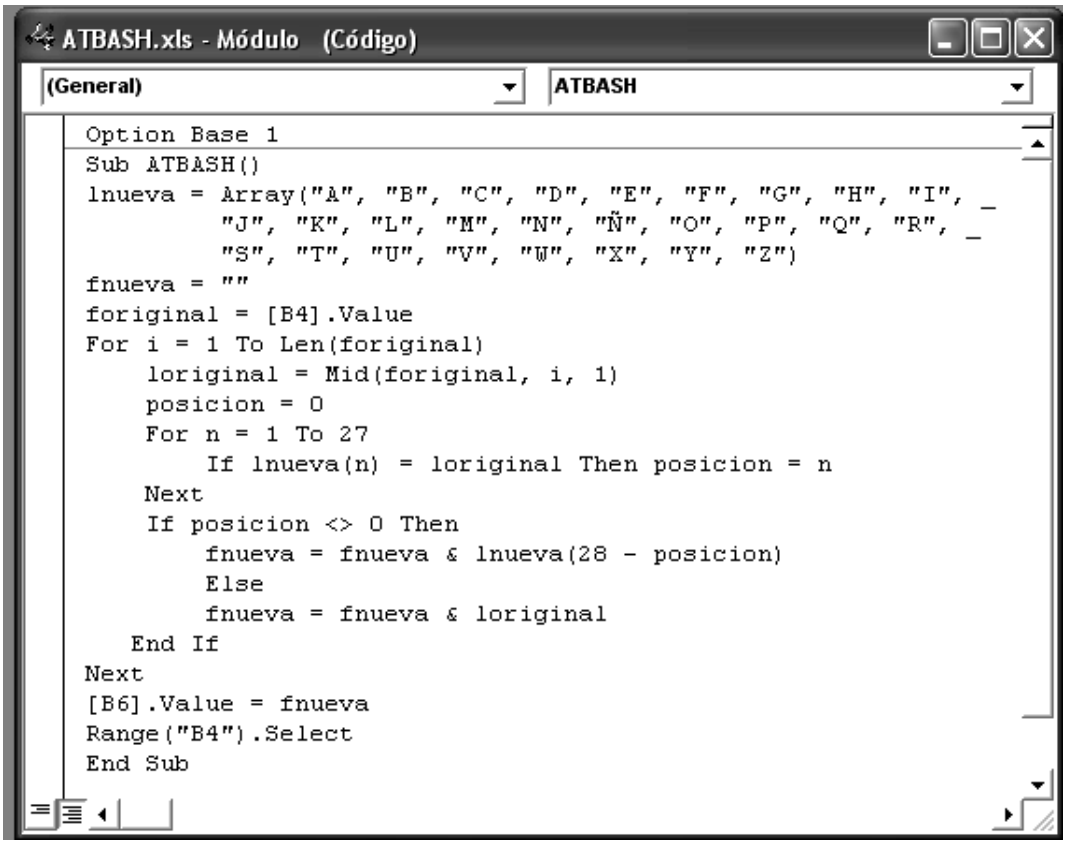
Vemos que la suma de ambas posiciones es 28.
Por eso, para calcular la posición de la nueva letra, solo tenemos que restarle a 28 la posición de la letra original.

Por ejemplo:

La letra de posición 12, se reemplaza por la de posición 28-12=16
La letra de posición 20, se reemplaza por la de posición 28-20=8

COMIENZA EL PROGRAMA

El código del módulo del programa Atbash es el que muestra la siguiente pantalla:



Comenzamos el programa cargando en una variable tipo vector las 27 letras del alfabeto. A es variable la llamamos **lnueva** (‘letra nueva’, ya que de ese vector vamos a sacar las letras necesarias para reemplazar las letras del mensaje original).

```
lnueva = Array("A", "B", "C", "D", "E", "F", "G", "H", "I", _
         "J", "K", "L", "M", "N", "Ñ", "O", "P", "Q", "R", _
         "S", "T", "U", "V", "W", "X", "Y", "Z")
```

Cada vez que queremos referirnos a algún elemento del vector escribimos: lnueva(n). Donde n indica la posición.

Normalmente, los subíndices de los vectores, comienzan en 0. Es decir, el rango de n va de 0 a 26, dado que son 27 letras.

Si queremos que en lugar de comenzar en cero, lo haga en 1, tenemos que indicárselo a Excel. Para eso escribimos al comienzo del módulo la instrucción Option base 1.

```
Option Base 1
```

Así queda el rango de n comenzando en 1.

Luego se inicializa una variable **fnueva** (‘frase nueva’), donde vamos a ir guardando el mensaje cifrado:

```
fnueva = " "
```

El mensaje original que queremos encriptar los vamos a guardar en la variable llamada **foriginal** (‘frase original’).

```
foriginal = [B4].Value
```

Esta instrucción le indica al programa que debe guardar en la variable foriginal el contenido de la celda B4. En esa celda va a estar escrito nuestro mensaje original.

Recorriendo el mensaje

```
For i = 1 To Len(foriginal)
.
.
.
Next
```

Este ciclo For es el encargado de recorrer la frase completa. Comenzando en la posición 1 y llegando hasta la última posición.

La instrucción Len le dice al programa cuantos caracteres tiene el mensaje.

```
loriginal = Mid(foriginal, i, 1)
```

Esta instrucción va tomando de a una las letras del mensaje.

Mid permite tomar parte de un texto. Tiene tres parámetros. El primero debe ser el texto, el segundo es la posición donde comienza la parte del texto que queremos tomar, el tercero es la cantidad de caracteres que necesitamos tomar a partir de la posición anterior.

En este caso, se va parando sucesivamente en todas las posiciones, i, y va tomando un sólo carácter. Lo va guardando sucesivamente en la variable loriginal (`letra original´).

```
posicion = 0
```

Esta variable llamada posicion nos permite ir guardando las posiciones de las letras del vector que vayamos necesitando. La inicializamos en cero.

```
For n = 1 To 27
   If lnueva(n) = loriginal Then posicion = n
Next
```

Este ciclo For recorre todo el vector. En ese recorrido va comparando la letra original con todas las letras del vector, y cuando la encuentra, guarda la posición, que es el subíndice del vector, en la variable posicion.

```
If posicion <> 0 Then
   fnueva = fnueva & lnueva(28 - posicion)
Else
  fnueva = fnueva & loriginal
End If
```

La variable **fnueva** (‘frase nueva’) va armando el mensaje encriptado concatenando sucesivamente las nuevas letras obtenidas del vector.

Conociendo la posición de la letra original, que está guardada en la variable **posicion**, obtiene la posición de la nueva letra restándola de 28. Luego busca en el vector cuál es la letra que ocupa esa posición: **lnueva(28 - posicion**).

El símbolo & le indica al programa que debe concatenar el contenido de ambas variables, **fnueva** y **lnueva.**

Puede suceder que el mensaje original tenga otros caracteres que no sean letras mayúsculas. Por ejemplo, comas, espacios, puntos, comillas, números, etc. En este programa, esos caracteres van a quedar iguales. Solo serán cambiadas las letras mayúsculas.

La instrucción **If** pregunta qué tipo de carácter contiene la variable **loriginal**. Si es una letra mayúscula, concatena la variable **fnueva** con la variable **lnueva** tomada del vector. Si se trata de cualquier otro carácter, concatena la variable **fnueva** directamente con **loriginal**. Así, cualquier otro carácter que contenga el mensaje original quedará sin modificar en el mensaje cifrado.

```
[B6].Value = fnueva
```

Por último coloca en la celda B6 de la planilla el valor de la variable **fnueva** que contiene el mensaje ya encriptado.

```
Range("B4").Select
```

Esta instrucción simplemente indica al cursor que se posicione en la celda B4.

NOTAS

Nuestro mensaje deberá escribirse en la celda B4 y el programa nos devolverá el mensaje encriptado en la celda B6.

Como Atbash es un método simétrico, del mismo modo que encriptamos, podemos desencriptar. Así, si hacemos al revés y colocamos el mensaje cifrado en la celda B4, obtendremos el mensaje original en la celda B6.

Es decir que el mismo programa nos sirve tanto para encriptar como para desencriptar mensajes.

Proponemos para el final, descifrar las palabras de un célebre científico italiano. Vale la pena. ¡Suerte!

VO FNREVIHL VHGZ VHXIRGL VN VO OVNTFZQV WV OZH ÑZGVÑZGRXZH HRVNWL HFH XZIZXGVIVH GIRZNTFOLH, XRIXFOLH B LGIZH URTFIZH TVLÑVGIRXZH, HRN OZH XFZOVH VH SFÑZNZÑVNGV RÑKLHRYOV XLÑKIVNWVI FNZ HLOZ KZOZYIZ. HRN VOOLH HLOL HV XLNHVTFRIZ EZTZI KLI FN LHXFIL OZYVIRNGL.
TZOROVL TZOROVR

COMENTARIOS FINALES

Esta propuesta tiene especial interés porque integra conocimientos matemáticos con técnicas de programación. Además, la referencia a *El escarabajo de oro* puede servir para acercar a los alumnos a la literatura de Edgar Allan Poe. Es adecuada para implementarse a partir del cuarto año de enseñanza media (segundo año del ciclo polimodal).

Desde el punto de vista matemático se aprovechan nociones tales como el sistema de numeración binario, permutaciones, cálculo de factoriales o los operadores lógicos.

El programa de encriptación propuesto está desarrollado bajo Excel ya que son varias las razones que lo hacen una herramienta ideal para acercar a los alumnos a la programación. Excel está disponible en todo laboratorio de computación en las escuelas y es muy fácil acceder a sus módulos de programación. Además permite desarrollar desde programas sencillos, como por ejemplo, uno que simplemente escriba un nombre y lo coloree, hasta otros muy complejos.

A partir de este primer contacto con la programación en Excel se podrán desarrollar muchos otros temas de matemática. Se pueden simular por ejemplo tiradas de una moneda para el calculo de probabilidades; programar diversas formas de realizar sorteos; juegos, como el Master Mind; seguimientos de campeonatos de fútbol; simular encuestas y graficar resultados; listar números primos; calcular cifras decimales de Pi; simular otros fenómenos aleatorios por el método de Montecarlo; y más…

REFERENCIAS BIBLIOGRÁFICAS

Encriptación, Rolando Bardelli, revista Users Nº122, Junio 2001.
Enciclopedia Microsoft Encarta 2000.
Proyectos con macros en Excel, Claudio Sánchez, MP Ediciones, 2002.
El escarabajo de oro, Edgar Allan Poe, Narraciones Extraordinarias, Editorial Salvat, Buenos Aires, 1969.

NOTA
La traducción de la frase encriptada es:
EL UNIVERSO ESTA ESCRITO EN EL LENGUAJE DE LAS MATEMATICAS SIENDO SUS CARACTERES TRIANGULOS, CIRCULOS Y OTRAS FIGURAS GEOMETRICAS, SIN LAS CUALES ES HUMANAMENTE IMPOSIBLE COMPRENDER UNA SOLA PALABRA. SIN ELLOS SOLO SE CONSEGUIRA VAGAR POR UN OSCURO LABERINTO.

GALILEO GALILEI