Tecnológico de Costa Rica

Escuela de Ingeniería en Computación

IC-1801 Taller de Programación

Prof. Mauricio Avilés

Proyecto Programado 1 – Criptografía y cifrado

Motivación

La criptografía es la técnica, ciencia o arte de la escritura secreta. El principio básico de la criptografía es mantener la privacidad de la comunicación entre dos personas, alterando el mensaje original de modo que sea incomprensible a toda persona distinta del destinatario.

Se puede decir que la criptografía es tan antigua como la civilización, cuestiones militares, religiosas o comerciales impulsaron desde tiempos remotos el uso de escrituras secretas. Los antiguos *egipcios* usaron métodos criptográficos, mientras el pueblo utilizaba la lengua demótica, los sacerdotes usaban la escritura hierática (jeroglífica) incomprensible para el resto. Los antiguos *babilonios* también utilizaron métodos criptográficos en su escritura cuneiforme.

En la actualidad existen diversos usos de técnicas criptográficas, se han extendido desde las redes de computadoras hasta el almacenamiento de la información. Algunos de los métodos básicos serán implementados como parte de esta tarea programada.

Cifrado por transposición

Este es un tipo de cifrado donde las unidades que conforman las palabras (letras, sílabas u otros) se cambian de orden según algún esquema. El caso que se debe implementar consiste en escribir las palabras del mensaje con sus letras en orden invertido.

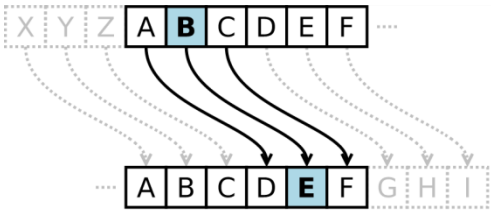
Ejemplo:

Texto: `es hora de buscar lo esencial`

Texto codificado: `se aroh ed racsub ol laicnese`

Cifrado César

El cifrado César, también conocido como cifrado por desplazamiento, es una técnica de codificación muy simple que fue utilizada por el dictador romano Julio César para comunicarse con sus generales. Consiste en sustituir cada letra del texto original, reemplazándola por otra letra que se encuentre un número fijo de posiciones más adelante en el alfabeto.



Ejemplo con desplazamiento = 3:

Texto: `espero tus problemas se acaben`

Texto codificado: `hvshur wxv sureñhodv vh dfdehp`

El mensaje cifrado se consigue adelantando tres letras cada una de las letras del mensaje, sin cambiar los espacios en blanco. Si la letra resultante es mayor que la Z, entonces debe continuarse al inicio del abecedario.

Si se desea decodificar, entonces debe repetirse el proceso con cada letra, pero retrocediendo tres posiciones cada una.

Sólo se modifican las letras del mensaje; números y signos de puntuación deben permanecer sin alteraciones. El desplazamiento a utilizar es variable, puede usarse cualquier valor entero entre 1 y la cantidad de letras del abecedario.

Codificación binaria

Se realiza sustitución de cada letra del alfabeto por un código binario de cinco dígitos. Los valores de las letras se escriben separados por un espacio en blanco y el espacio en blanco se traduce como un asterisco (*) entre las palabras. La tabla de equivalencias a utilizar es la siguiente.

a	00000	j	01001	r	10010
b	00001	k	01010	s	10011
c	00010	l	01011	t	10100
d	00011	m	01100	u	10101
e	00100	n	01101	v	10110
f	00101	ñ	01110	w	10111
g	00110	o	01111	x	11000
h	00111	p	10000	y	11001
i	01000	q	10001	z	11010

Ejemplo:

Texto: `dobla tu energia en partes iguales`

Texto codificado:

```
00011 01111 00001 01011 00000 * 10100 10101 * 00100 01101 00100 10010
00110 01000 00000 * 00100 01101 * 10000 00000 10010 10100 00100 10011 *
01000 00110 10101 00000 01011 00100 10011
```

1337

El alfabeto 1337, también conocido como *leet*, *eleet* o *leetspeak*, es un alfabeto alternativo que era utilizado principalmente en internet. Utiliza varias combinaciones de caracteres para remplazar las letras del alfabeto latino. Por ejemplo, la palabra leet puede escribirse como 1337, l33t, 31337 o 3l33t.

El término deriva de la palabra élite y se originó en los *bulletin board systems* de los años 80, donde tener el estatus de “élite” permitía a los usuarios acceder archivos, juegos y chats especiales. En un inicio era conocido y utilizado por *hackers* y *crackers* que intentaban hablar en clave para no ser detectados fácilmente. Desde entonces el 1337 ha sido adoptado popularmente. Actualmente se utiliza con fines humorísticos o como una forma nostálgica de recordar los inicios de internet.

Aunque no existen lineamientos estrictos establecidos sobre su uso, hay ciertas reglas o convenciones que permiten que el 1337 sea comprendido por cualquiera. El 1337 no solamente se limita a sustituir caracteres, si no que también existen palabras y expresiones propias del 1337; pero para esta asignación se va a limitar la codificación a la sustitución de letras.

La siguiente tabla muestra algunas de las equivalencias utilizadas para cada letra del abecedario.

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
4 @ /-\ /\ ^	8 |3 6 13 |3]3	(< ¢ { ©	|) [)]) I> |> cl	3 & € [-	|=]= } ph (=	6 (_+	|-| #]-[[-])-((-) :-: }{ }-{	! 1 | ¡	_| _/] </ _)	|< |X |{	1 |_ | |_ ¬	44 /\/\ |\/| |v| IYI IVI [V] ^^ nn //\\//\\ (V) (\/) /|\ /|/| .\\ /^^\ /V\ |^^|	|\\| /\/ //\\// [\] <\> {\} // []\[]]\[~	0 () [] ¤	|* |o |º |> |" ¶	0_ 0, (,) <| 9	|2 /2 I2 |^ |~ ® |2 [z l2	5 $ z §	7 + -|-	|_| (_) µ [_] _/ _\ /_/	\/ \\//	\/\/ vv '// \\' \^/ (n) \X/ \\|/ _|_/ \\//\\// _:_/]I[UU	% >< }{ ×)(	`/ `(-/ '/	2 ~/_ 7_

Ejemplo:

Texto: `todo va a estar mas o menos bien`

Texto codificado: `70|)0 \/4 4 3574|2 |\/|45 0 |\/|3|\|05 813|\|`

Para la implementación de esta codificación cada letra debe convertirse a alguno de los equivalentes mostrados en la tabla. La escogencia del equivalente a utilizar debe ser aleatoria. En esta codificación existe libertad de agregar más equivalencias, siempre y cuando sea posible deducir cuál es la letra que se está codificando. No se implementará la decodificación del mensaje por razones de complejidad.

Cifrado monoalfabético con palabra clave

En este cifrado se mapea cada letra con otra letra del alfabeto. El orden del alfabeto cifrado se obtiene a partir de una palabra clave. Supongamos que se utiliza la palabra clave "CONFIGURABLES"

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

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

Las letras (sin repetir) de la palabra clave se usan como inicio del alfabeto y el resto del alfabeto utiliza las letras que faltan en el mismo orden en que aparecen en el alfabeto normal.

Texto: `formemos una banda de rocanrol`

Texto codificado: `gjñsisjp tdc ocdfc fi ñjncdñje`

Cifrado Vigenère

Es un cifrado que utiliza una palabra clave para realizar el proceso de cifrado. Cada palabra se procesa de forma independiente. Para realizar el proceso se utiliza la siguiente correspondencia de valores:

a	0	f	5	k	10	o	15	t	20	y	25
b	1	g	6	l	11	p	16	u	21	z	26
c	2	h	7	m	12	q	17	v	22		
d	3	i	8	n	13	r	18	w	23		
e	4	j	9	ñ	14	s	19	x	24		

A la primera letra del texto a modificar se le suma el valor de la primera letra de la palabra clave, esto da el valor de la letra en el texto codificado, pero es necesario hacerle módulo 27 para que el código no se pase de las letras disponibles. Luego, para a siguiente letra, se le suma el valor de la siguiente letra de la palabra clave. Cuando se acaban los valores de la palabra clave, se repiten. Esto se hace hasta codificar todo el texto.

Ejemplo con palabra clave "amigo":

Texto: `hoy celebraremos como familia`

a	m	i	g	o
0	12	8	6	15

h	o	y		c	e	l	e	b	r	a	r	e	m	o	s		c	o	m	o		f	a	m	i	l	i	a
7	15	25		2	4	11	4	1	18	0	18	4	12	15	19		2	15	12	15		5	0	12	8	11	8	0
+																													
a	m	i		g	o	a	m	i	g	o	a	m	i	g	o		a	m	i	g		o	a	m	i	g	o	a
0	12	8		6	15	0	12	8	6	15	0	12	8	6	15		0	12	8	6		15	0	12	8	6	15	0
=																													
h	a	g		i	s	l	p	j	x	o	r	p	t	u	h		c	a	t	u		t	a	x	p	q	w	a
7	0	6		8	19	11	16	9	24	15	18	16	20	21	7		2	0	20	21		20	0	24	16	17	23	0

Texto codificado: `hag islpjxorptuh catu taxpqwa`

La codificación de las letras va desde 0 hasta 26. En algunos casos la suma de dos letras puede resultar en un código mayor que la Z, cuando esto suceda debe restar 27 al valor resultante (o utilizar la operación de módulo 27). Por otro lado, cuando se decodifica, hay que hacer el proceso inverso, restar los valores de las letras según la palabra clave utilizada.

Cifrado PlayFair modificado

Este cifrado consiste en una técnica de sustitución dinámica que también utiliza una palabra clave. Con la palabra clave se genera una matriz de letras que se usa para sustituir las letras de dos en dos, en vez de individualmente.

Supongamos la palabra clave "PROFUNDIZASTE". Con esta palabra se genera una matriz de seis filas y cinco columnas con todas las letras del abecedario. Si la palabra clave tuviera letras repetidas, deben eliminarse.

```
P R O F U
N D I Z A
S T E B C
G H J K L
M Ñ Q V W
X Y 1 2 3
```

Dado que la cantidad de letras no alcanza para completar la matriz, utilizaremos los números 1, 2 y 3 para completarla.

Ahora supongamos que la palabra a codificar es: MURCIELAGOS. Si la palabra a codificar tuviera alguna letra repetida consecutivamente (por ejemplo "ACCION"), las separamos con un 1 ("AC1CION"). Luego, hay que hacer es agrupar las letras en parejas: MU RC IE LA GO S1. Como la cantidad de letras es impar, entonces la última letra la agrupamos con el número 1.

Seguidamente procedemos a codificar cada par de letras. Para cada par de letras pueden darse tres casos, y cada caso hacemos la codificación de forma diferente.

1. Están en diferente fila, diferente columna

   Cada letra se sustituye por la que se encuentra en la misma fila y en la columna de la otra letra.

2. Misma fila, diferente columna

   Cada letra se sustituye por la que se encuentra en la posición contigua a la derecha. Si la letra está en la columna de más a la derecha, se sustituye por la que está en la primera columna.

3. Diferente fila, misma columna

   Cada letra se sustituye por la que se encuentra en la posición contigua hacia abajo. Si la letra está en la última fila, se sustituye por la que está en la primera fila.

Continuando con el ejemplo:

MU→WP (Caso 1)

RC→UT (Caso 1)

IE→EJ (Caso 3)

LA→CW (Caso 3)

GO→JP (Caso 3)

S1→EX (Caso 1)

Texto codificado: WPUTEJCWJPEX

Otro ejemplo, codificar la palabra: PROFESSOR. Primero se separan las letras repetidas contiguas: PROFES1SOR. Se separa en parejas y se completa con un 1 si la cantidad es impar: PR OF FE S1 SO R1.

PR→RO (Caso 2)

OF→FU (Caso 2)

FE→OB (Caso 1)

S1→EX (Caso 1)

SO→EP (Caso 1)

R1→OY (Caso 1)

Texto codificado: ROFUOBEXEPOY

La decodificación de este método se realiza casi de la misma forma que la codificación, pero los casos 2 y 3 deben hacerse restando posiciones a las filas y columnas en vez de sumar (en sentido contrario).

Software a desarrollar

El objetivo de esta tarea es programar todos los métodos de codificación y decodificación. Únicamente el método 1337 no va requerir método de decodificación por razones de complejidad.

Para todos los casos se trabajará con el siguiente alfabeto: 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 y el espacio en blanco. Al texto a codificar debe eliminársele las tildes y mayúsculas.

Deben implementarse las funciones necesarias en Python para cumplir con los requerimientos de esta tarea. Las funciones deben ser las siguientes:

1. transposicionCod(texto) → codifica texto usando transposición
2. transposicionDec(texto) → decodifica texto usando transposición
3. cesarCod(texto, desplazamiento) → codifica usando cifrado césar con el desplazamiento indicado

4. cesarDec(texto, desplazamiento) → decodifica usando cifrado césar con el desplazamiento indicado
5. binarioCod(texto) → codifica el texto usando cifrado binario
6. binarioDec(texto) → decodifica el texto usando cifrado binario
7. leetCod(texto) → codifica el texto usando cifrado 1337
8. monoCod(texto, palabra) → codifica usando cifrado monoalfabético con la palabra clave indicada
9. monoDec(texto, palabra) → decodifica usando cifrado monoalfabético con la palabra clave indicada
10. vigenereCod(texto, palabra) → codifica el texto usando cifrado vigenere con la palabra clave indicada
11. vigenerDec(texto, palabra) → decodifica el texto usando cifrado vigenere con la palabra clave indicada
12. playfairCod(texto, palabra) → codifica el texto usando cifrado playfair con la palabra clave indicada
13. playfairDec(texto, palabra) → decodifica el texto usando la palabra clave indicada

Documentación

Toda función debe llevar como documentación interna comentarios con lo siguiente:

- Descripción de la función
- Entradas
- Salidas
- Restricciones

En cuanto a la documentación externa, debe entregarse un manual de usuario en formato PDF que explique a cualquier persona cómo utilizar las funciones en el *shell* de Python.

Entrega

El tiempo asignado para la tarea programada es de 3 semanas. El trabajo se realizará en parejas.

La entrega debe hacerse en la sección de evaluaciones del curso en el TEC-Digital. Debe entregarse un archivo comprimido .ZIP con el archivo .py de la solución y un archivo .PDF con la documentación.

Evaluación

La tarea tiene un valor de 20% de la nota final, en el rubro de Proyectos Programados.

Desglose de la evaluación de la tarea programada:

Documentación:	20%
Programación:	80%

Recomendaciones adicionales

Antes de escribir código entienda muy bien el problema y trate de diseñar los algoritmos a utilizar. Esto mediante un análisis donde tome el problema y lo subdivide en problemas más pequeños y fáciles de solucionar, como se ha visto en clases.

Piense en las acciones que un usuario podría hacer para causar fallos en las funciones y tome las medidas necesarias. Por ejemplo, que pasaría si a un método se le introducen parámetros de otros tipos diferentes a los aceptados.

Si bien es cierto que lo más importante es que el programa funcione, no descuide la documentación, ya que una mala documentación podría causar que su sistema sea malinterpretado, obteniendo una nota que no merezca.

Cualquier actividad fraudulenta será procesada según el Reglamento de Enseñanza-Aprendizaje del Instituto Tecnológico de Costa Rica.