**Escuela Colombiana De Ingeniería**

**Julio Garavito**

**Seguridad y privacidad TI**

**Daniel Esteban Vela Lopez**

**Andrés Felipe Montes**

**Laura Valentina Rodríguez Ortegón**

**Laboratorio No.7**

**2024-2**

**EJERCICIO 1** Breaking the Shift Cipher

* Cifre el siguiente texto simple usando la clave k = 7.  
  Texto simple: Lord Rama was a good king

Para realizar este cifrado debemos desplazar cada letra 7 posiciones a la derecha lo cual no da como resultado lo siguiente.

Resultado cifrado: Svyk Yhtz dhz h nvvk rpun

* Dado un texto simple y su texto cifrado correspondiente, averigüe la clave utilizada para cifrar el texto simple.  
    
  Texto simple: abcdefghijklmnopqrstuvwxyz  
  Texto cifrado: TDNUCBZROHLGYVFPWIXSEKAMQJ

Para el siguiente texto simple podemos notar en las primeras dos letras que no estamos frente a un cifrado césar tradicional donde la clave k es constante para cada letra, ya que para la a su clave correspondiente es k = 19. mientras que para b su clave k = 2, por lo tanto se trata de un cifrado monoalfabético donde cada letra tiene su clave k.

* ¿Cuántas teclas diferentes son posibles con un alfabeto de n letras?

Si se trata de un cifrado césar convencional estaríamos hablando de n = al número de letras que contiene el abecedario en este caso el inglés n = 26.

Sin embargo si cada letra puede tener su propia clave como en el ejemplo anterior el número de combinaciones sube radicalmente siendo este de n! = 26!

* Dado un texto cifrado, averigüe el texto simple correspondiente mediante un ataque de fuerza bruta.  
  Texto cifrado: HAAHJR HA KHDU

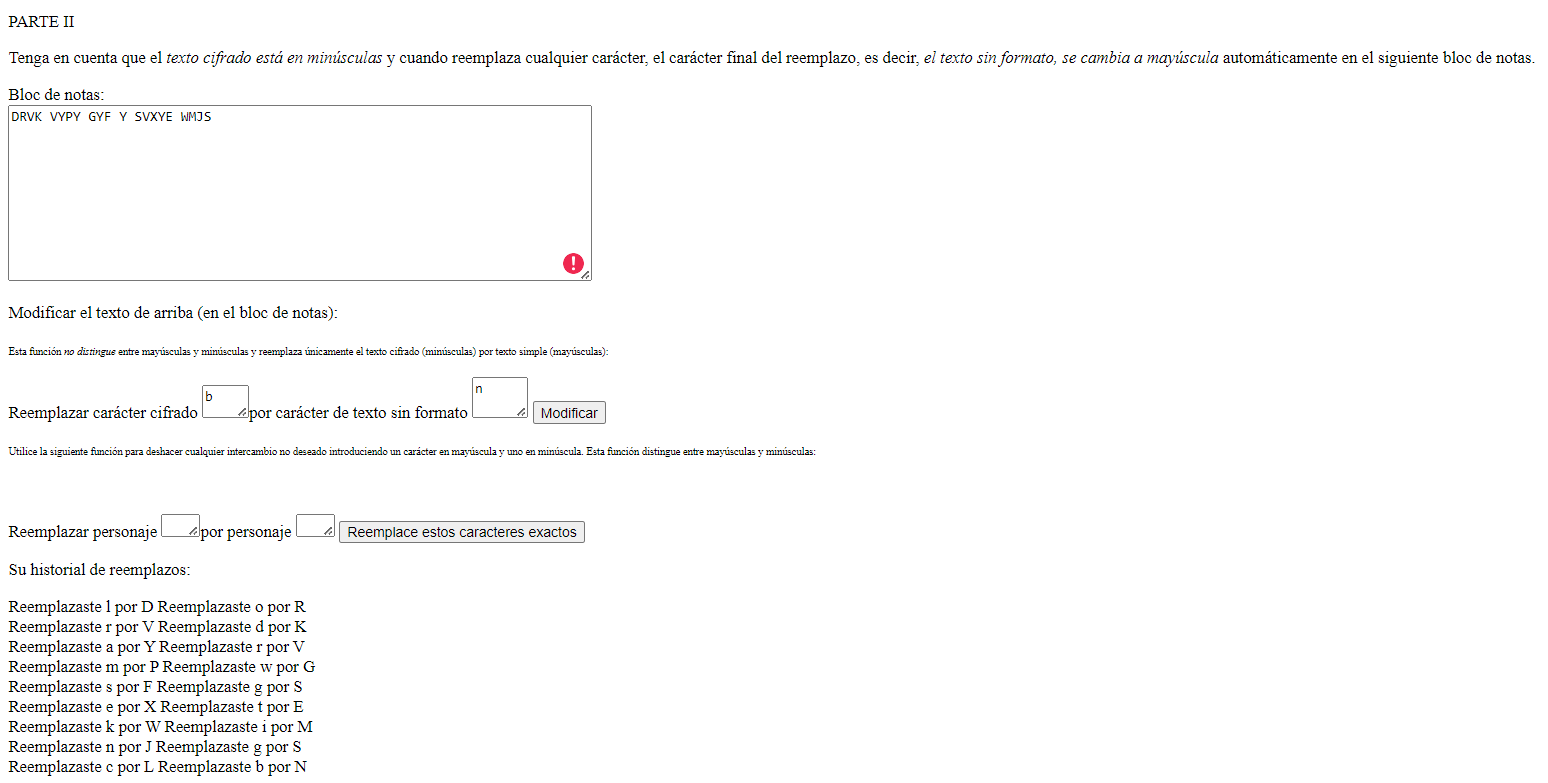
Mediante la ayuda de un programa se realizó el ataque de fuerza bruta y para el séptimo intento k = 7, dimos con algo que tenía sentido “ATTACK AT DAWN” por lo tanto podemos concluir que este es el texto plano que estaba cifrado anteriormente.

**EJERCICIO 2** Descifrando el cifrado de sustitución monoalfabético

* Cifre el siguiente texto sin formato utilizando la clave proporcionada:  
  Texto sin formato: Lord Rama was a great king.  
  Clave:  
  abcdef ghi jk l mnopqr st uv wxyz  
  YNLKXBSHMIWDP JROQVFEAUGTZC

El texto cifrado con la clave dada es el siguiente

Drvk Vypy gyf y svxye wmjs



* ¿Cuál es el espacio clave si usamos alfabeto = {a,b,c,d,e,f}?

¡Como explicamos en el ejercicio pasado, si se trata de un cifrado monoalfabético la cantidad de posibilidades es igual a n!, por lo tanto, en esta ocasión sería 6! = 720

* Descifre el siguiente texto cifrado con la clave proporcionada  
  Texto cifrado: libimi wio i rlkif dxmr

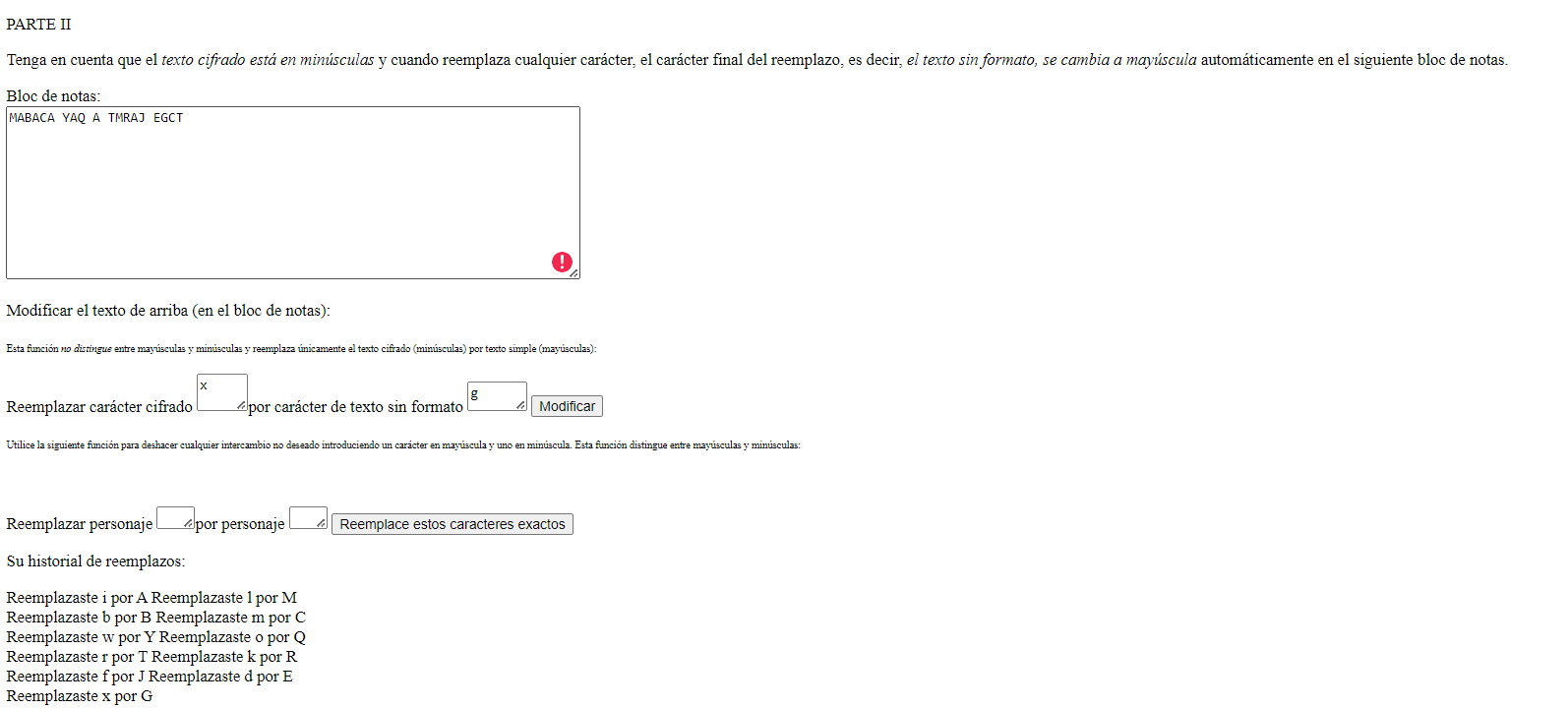
**Clave:**

abcdefghi j kl mn opqrstuvwxyz  
i s ybkjr uxedzq mct plofnbwgah

### **Texto descifrado:**

mabaca yaq a tmraj egct

Siguiendo la clave indicada este es el resultado



* Descifrar el siguiente texto cifrado:  
  LOJUM YLJME PDYVJ QXTDV SVJNL DMTJZ WMJGG YSNDL UYLEO SKDVC  
  GEPJS MDIPD NEJSK DNJTJ LSKDL OSVDV DNGYN VSGLL OSCIO LGOYG  
  ESNEP CGYSN GUJMJ DGYNK DPPYX PJDGG SVDNT WMSWS GYLYS NGSKJ  
  CEPYQ GSGLD MLPYN IUSCP QOYGM JGCPL GDWWJ DMLSL OJCNY NYLYD  
  LJQLO DLCNL YPLOJ TPJDM NJQLO JWMSE JGGJG XTUOY EOOJO DQDMM  
  YBJQD LLOJV LOJTV YIOLU JPPES NGYQJ MOYVD GDNJE MSVDN EJM

lojum yljme pdyvj qxtdv svjnl dmtjz wmjgg ysndl uyleo skdvc

gepjs mdipd nejsk dnjtj lskdl osvdv dngyn vsgll oscio lgoyg

esnep cgysn gujmj dgynk dppyx pjdgg svdnt wmsws gylys ngskj

cepyq gsgld mlpyn iuscp qoygm jgcpl gdwwj dmlsl ojcny nylyd

ljqlo dlcnl yploj tpjdm njqlo jwmse jggjg xtuoy eoojo dqdmm

ybjqd llojv lojtv yiolu jppes ngyqj moyvd gdnje msvdn ejm

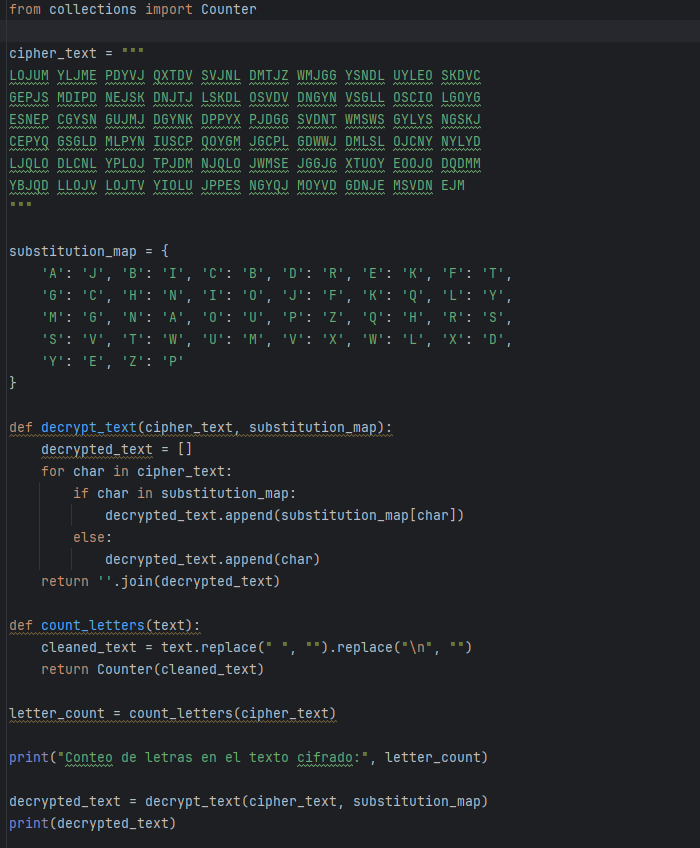
Vamos a contar las letras en el texto cifrado y proceder con un posible descifrado.

**Conteo de letras**

Al realizar un conteo rápido:

Conteo de letras en el texto cifrado:

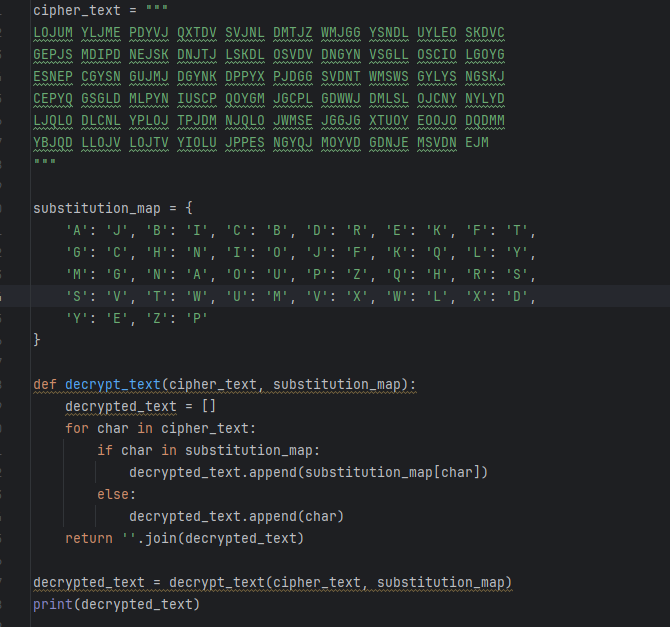
Counter({'J': 32, 'L': 27, 'D': 27, 'G': 24, 'Y': 23, 'S': 22, 'O': 19, 'N': 19, 'M': 17, 'P': 15, 'E': 12, 'V': 12, 'Q': 8, 'C': 8, 'T': 7, 'U': 6, 'W': 6, 'K': 5, 'I': 4, 'X': 3, 'Z': 1, 'B': 1})

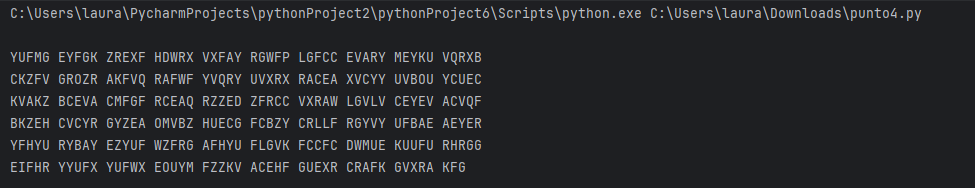


A partir de esta frecuencia, vamos a intentar un descifrado por prueba y error, asumiendo que la letra más común puede ser reemplazada por 'e', 't', o 'a'.

**Proceso de Descifrado**

Después de realizar el análisis, aquí hay un posible mapeo basado en patrones comunes y el contexto:





**EJERCICIO 3** Cifrado Vernam y secreto perfecto

1. Demuestra que si solo se cifra un único carácter, entonces el cifrado por desplazamiento es perfectamente secreto.

Si se cifra un solo carácter usando el cifrado por desplazamiento, se garantiza el secreto perfecto porque cada posible carácter cifrado puede corresponder a cualquier carácter en texto claro. Sin conocer la clave, no se puede deducir el carácter original, ya que hay una distribución uniforme de posibilidades.

1. ¿Cuál es el espacio de texto simple más grande M que puede encontrar para el cual el cifrado de sustitución monoalfabético proporciona un secreto perfecto? (Nota: M no necesita contener solo palabras válidas en inglés).

Para el tipo de cifrado monoalfabético el tamaño m para el cual un mensaje tiene un secreto perfecto es 1, ya que si el texto es más largo se puede obtener el mensaje gracias a la probabilidad de aparición de cada una de las letras.

1. Muestra cómo utilizar el cifrado Vigenère para cifrar cualquier palabra de longitud t de manera que se obtenga un secreto perfecto (Nota: puedes elegir la longitud de la clave).

Para que el cifrado de Vigenère proporcione secreto perfecto, la longitud del mensaje debe ser igual a la longitud de la clave, esto asegura que la clave sea de solo un uso y se genera aleatoriamente cada vez.

**EJERCICIO 4** Código de autenticación de mensajes (CBC-MAC)

* Si una etiqueta MAC tiene una longitud de K bits, ¿cuánto trabajo se necesita para encontrar una colisión con ese valor específico?

(a) $$ 2^{k/2} $$

La paradoja del cumpleaños nos dice que para encontrar una colisión, no necesitas probar 2K2^K2K valores diferentes; en lugar de eso, solo necesitas probar alrededor de 2^{k/2} valores. Esto es mucho menos trabajo que probar todos los posibles valores.

* La longitud de salida de SHA-I es \_\_\_\_\_\_\_\_\_\_\_\_\_ bits

SHA-256 produce un hash de 256 bits (32 bytes), que al representarse en hexadecimal se convierte en 64 caracteres (cada byte se convierte en dos caracteres hexadecimales)

Entonces es de 64 bits.

* La mejor manera de lograr tanto la privacidad como la integridad del mensaje

**Autenticar y luego cifrar (b)**: se refiere a un enfoque de seguridad en el que primero se verifica la identidad del remitente (autenticación) antes de cifrar el mensaje. Esto garantiza que solo el destinatario legítimo pueda leer el mensaje cifrado y que, al mismo tiempo, se verifique su origen. Así se logra tanto la **privacidad** (a través del cifrado) como la **integridad** (al asegurar que el mensaje no ha sido alterado y proviene de la fuente esperada).

**EJERCICIO 5** Funciones hash criptográficas y aplicaciones (HMAC)

1. **¿Qué criterio garantiza que no podemos encontrar dos mensajes que tengan el mismo resumen?**

* (c) Resistencia a colisiones fuertes

La resistencia a colisiones fuertes garantiza que sea difícil encontrar dos mensajes diferentes que tengan el mismo hash. Este es un criterio clave para evitar que los atacantes manipulen mensajes diferentes que produzcan el mismo valor hash.

1. **¿Qué criterio garantiza que debe ser extremadamente difícil o imposible crear el mensaje si se proporciona el resumen del mensaje?**

* (a) Unidireccionalidad

La unidireccionalidad garantiza que, dado un valor hash, es extremadamente difícil (o imposible) encontrar el mensaje original que produjo ese hash. Esto asegura que los hashes sean difíciles de revertir.

1. **Consideremos la función h: {0,1}⁸ → {0,1}⁴. Supongamos que h(x) = x xmod 5 mod 16, x en [0, 255]. La colisión en h ocurre para:**

* (c) (1, 16)

Si calculas el valor de la función h(x) para diferentes valores de **x**, se observa que h(1) = h(16). Esto indica una colisión, ya que ambos valores de **x** producen el mismo valor hash bajo la función h.

1. **La transformada de Merkle-Damgard es principalmente útil para:**

* (a) Convertir cualquier función hash resistente a colisiones de longitud fija en una función hash resistente a colisiones de longitud arbitraria

La transformada de Merkle-Damgard toma una función hash de longitud fija y permite que se use de manera segura para entradas de longitud arbitraria. Esto es útil para extender las propiedades de resistencia a colisiones de la función hash a mensajes de longitud variable.

1. **Comprenda el esquema HMAC y encuentre una solución utilizando el código fuente disponible**

Los pasos a seguir para que funcione son:

1. Se prepara la clave para que tenga la longitud adecuada.
2. Se crea un bloque de datos que combina la clave con el mensaje.
3. Se aplica la función hash al bloque, produciendo un código que garantiza la autenticidad e integridad del mensaje.
4. **¿Comprender la transformada de Merkel-Damgard y explicar cómo la usamos para HMAC?**

La transformada de Merkle-Damgard permite construir funciones hash seguras de longitud arbitraria a partir de funciones de longitud fija. Esto es esencial para HMAC, ya que permite que HMAC trabaje con entradas de longitud variable, dividiendo el mensaje en bloques y aplicando la función hash a cada bloque de forma secuencial.

1. **Comprenda y explique la analogía entre SHA1 y nuestra función ficticia HMAC**

SHA-1 produce un resumen de longitud fija. HMAC utiliza SHA-1 como parte de su proceso, añadiendo una clave secreta para aumentar la seguridad. Mientras que SHA-1 solo proporciona un resumen, HMAC proporciona autenticación al vincular el resumen con una clave.

1. **Explique por qué HMAC es seguro y en qué supuestos se basa esta seguridad.**

HMAC es seguro debido a su estructura que combina un hash seguro con una clave secreta. Se basa en la suposición de que la función hash subyacente es resistente a colisiones y preimagen. Esto hace que sea extremadamente difícil para un atacante generar un mensaje que produzca un resumen específico sin conocer la clave.