UNIVERSIDAD NACIONAL DE SAN AGUSTÍN DE AREQUIPA

FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS

ESCUELA PROFESIONAL DE CIENCIA DE LA COMPUTACIÓN

SEGURIDAD EN COMPUTACIÓN

TEMA:

Funciones elementales de Criptografía

Docente:

Mg. Rolando Jesús Cárdenas Talavera

Semestre:

VIII

Presentado por:

Philco Puma, Josue Samuel

Arequipa - Perú

2025

Ejercicios de Laboratorio

Sobre el texto claro mostrado a continuación:

> **Hay golpes en la vida, tan fuertes... ¡Yo no sé!**
> **Golpes como del odio de Dios; como si ante ellos,**
> **la resaca de todo lo sufrido**
> **se empozara en el alma... ¡Yo no sé!**
> **Son pocos; pero son... Abren zanjas oscuras**
> **en el rostro más fiero y en el lomo más fuerte.**
> **Serán tal vez los potros de bárbaros Atilas;**
> **o los heraldos negros que nos manda la Muerte.**
> **Son las caídas hondas de los Cristos del alma**
> **de alguna fe adorable que el Destino blasfema.**
> **Esos golpes sangrientos son las crepitaciones**
> **de algún pan que en la puerta del horno se nos quema.**
> **Y el hombre... Pobre... ¡pobre! Vuelve los ojos, como**
> **cuando por sobre el hombro nos llama una palmada;**
> **vuelve los ojos locos, y todo lo vivido**
> **se empoza, como charco de culpa, en la mirada**
> **Hay golpes en la vida, tan fuertes... ¡Yo no sé!**

Implementar las siguientes operaciones de preprocesamiento, en cada caso debe mostrar el código de la solución y la salida parcial resultante en cada caso.

El entorno utilizado para el laboratorio es Google Colab en lenguaje de programación Python, primero crearemos una variable global que lea nuestro texto de entrada desde un archivo.

```python
name_file = "Text_input.txt"

with open(name_file, "r") as file:
  text = file.read()

print(text)
```

Con esta inicialización ya podemos trabajar los ejercicios:

1. **Realizar las siguientes sustituciones: j × i, h × i, ñ × n, k × l, u × v, w × v, y × z**

   Bueno, acá simplemente nos pide sustituir letras específicas, a esto se le conoce como normalización del alfabeto para reducir la variabilidad para un análisis posterior.

```python
sustitutions = {
    "j" : "i",
    "h" : "i",
```

```
    "ñ" : "n",
    "k" : "l",
    "u" : "v",
    "w" : "v",
    "y" : "z"
}

text_sustitution = text
for key, value in sustitutions.items():
  text_sustitution = text_sustitution.replace(key, value)
  text_sustitution = text_sustitution.replace(key.upper(), value.upper())

print(text_sustitution)
```

Acá creamos una lista con los valores que vamos a cambiar, y por cada valor que encuentre dentro de la lista lo reemplaza con su letra equivalente, la salida correspondiente es la siguiente:

```
Iaz golpes en la vida, tan fvertes... ¡Zo no sé!
Golpes como del odio de Dios; como si ante ellos,
la resaca de todo lo svfrido
se empozara en el alma... ¡Zo no sé!

Son pocos; pero son... Abren zanias oscvras
en el rostro más fiero z en el lomo más fverte.
Serán tal vez los potros de bárbaros Atilas;
o los ieraldos negros qve nos manda la Mverte.

Son las caídas iondas de los Cristos del alma
de algvna fe adorable qve el Destino blasfema.
Esos golpes sangrientos son las crepitaciones
de algún pan qve en la pverta del iorno se nos qvema.

Z el iombre... Pobre... ¡pobre! Vvelve los oios, como
cvando por sobre el iombro nos llama vna palmada;
vvelve los oios locos, z todo lo vivido
se empoza, como ciarco de cvlpa, en la mirada

Iaz golpes en la vida, tan fvertes... ¡Zo no sé!
```

2. Elimine las tildes

Ahora procedemos a eliminar las tildes para que los carácteres con tildes sean tratadas como sus versiones simples unificando el conteo de carácteres.

```python
import unicodedata

def non_accent(text):
  text_modify = unicodedata.normalize("NFD", text)
  return text_modify.encode("ascii", "ignore").decode("utf-8")
```

```
text_non_accent = non_accent(text_sustitution)
print(text_non_accent)
```

En esta parte debemos eliminar las tildes del texto, lo podemos realizar con una librería de Python que es **unicodedata** e ignorar las tildes encontradas. La salida es la siguiente:

```
Iaz golpes en la vida, tan fvertes... Zo no se!
Golpes como del odio de Dios; como si ante ellos,
la resaca de todo lo svfrido
se empozara en el alma... Zo no se!

Son pocos; pero son... Abren zanias oscvras
en el rostro mas fiero z en el lomo mas fverte.
Seran tal vez los potros de barbaros Atilas;
o los ieraldos negros qve nos manda la Mverte.

Son las caidas iondas de los Cristos del alma
de algvna fe adorable qve el Destino blasfema.
Esos golpes sangrientos son las crepitaciones
de algun pan qve en la pverta del iorno se nos qvema.

Z el iombre... Pobre... pobre! Vvelve los oios, como
cvando por sobre el iombro nos llama vna palmada;
vvelve los oios locos, z todo lo vivido
se empoza, como ciarco de cvlpa, en la mirada

Iaz golpes en la vida, tan fvertes... Zo no se!
```

3. Convierta todas las letras a mayúsculas

Con esto vamos a buscar que no haya distinción entre mayúsculas y minúsculas.

Python
```
def change_letters_to_upper(text):
  return text.upper()

text_upper = change_letters_to_upper(text_non_accent)
print(text_upper)
```

Acá simplemente usando la función **upper** podemos cambiar las letras minúsculas a mayúsculas.

```
IAZ GOLPES EN LA VIDA, TAN FVERTES... ZO NO SE!
GOLPES COMO DEL ODIO DE DIOS; COMO SI ANTE ELLOS,
LA RESACA DE TODO LO SVFRIDO
SE EMPOZARA EN EL ALMA... ZO NO SE!

SON POCOS; PERO SON... ABREN ZANIAS OSCVRAS
EN EL ROSTRO MAS FIERO Z EN EL LOMO MAS FVERTE.
SERAN TAL VEZ LOS POTROS DE BARBAROS ATILAS;
O LOS IERALDOS NEGROS QVE NOS MANDA LA MVERTE.

SON LAS CAIDAS IONDAS DE LOS CRISTOS DEL ALMA
DE ALGVNA FE ADORABLE QVE EL DESTINO BLASFEMA.
ESOS GOLPES SANGRIENTOS SON LAS CREPITACIONES
DE ALGUN PAN QVE EN LA PVERTA DEL IORNO SE NOS QVEMA.

Z EL IOMBRE... POBRE... POBRE! VVELVE LOS OIOS, COMO
CVANDO POR SOBRE EL IOMBRO NOS LLAMA VNA PALMADA;
VVELVE LOS OIOS LOCOS, Z TODO LO VIVIDO
SE EMPOZA, COMO CIARCO DE CVLPA, EN LA MIRADA

IAZ GOLPES EN LA VIDA, TAN FVERTES... ZO NO SE!
```

4. **Elimine los espacios en blanco y los signos de puntuación. Guarde el resultado en el archivo HERALDOSNEGROS_pre.txt**

   Acá eliminamos todos los espacios en blanco y signos de puntuación para que solo el texto sea de letras.

```python
import re

text_output = "HERALDOSNEGROS_pre.txt"

def delete_spaces_and_punctuation(text):
  text_modify = text.replace(" ", "")
  text_modify = text_modify.replace(".", "")
  text_modify = text_modify.replace(",", "")
  text_modify = text_modify.replace(";", "")
  text_modify = text_modify.replace(":", "")
  text_modify = text_modify.replace("!", "")
  text_modify = text_modify.replace("¡", "")
  return text_modify

text_delete_spaces_and_punctuation =
delete_spaces_and_punctuation(text_upper)
print(text_delete_spaces_and_punctuation)

with open(text_output, "w") as file:
  file.write(text_delete_spaces_and_punctuation)
```

   Acá debemos eliminar los signos de puntuación y los espacios en blanco, se puede usar la función **replace** para hacerlo, y también el resultado debemos almacenarlo en un nuevo archivo llamado **HERALDOSNEGROS_pre.txt**.

```
IAZGOLPESENLAVIDATANFVERTESZONOSE
GOLPESCOMODELODIODEDIOSCOMOSIANTEELLOS
LARESACADETODOLOSVFRIDO
SEEMPOZARAENELALMAZONOSE

SONPOCOSPEROSONABRENZANIASOSCVRAS
ENELROSTROMASFIEROZENELLOMOMASFVERTE
SERANTALVEZLOSPOTROSDEBARBAROSATILAS
OLOSIERALDOSNEGROSQVENOSMANDALAMVERTE

SONLASCAIDASIONDASDELOSCRISTOSDELALMA
DEALGVNAFEADORABLEQVEELDESTINOBLASFEMA
ESOSGOLPESSANGRIENTOSSONLASCREPITACIONES
DEALGUNPANQVEENLAPVERTADELIORNOSENOSQVEMA

ZELIOMBREPOBREPOBREVVELVELOSOIOSCOMO
CVANDOPORSOBREELIOMBRONOSLLAMAVNAPALMADA
VVELVELOSOIOSLOCOSZTODOLOVIVIDO
SEEMPOZACOMOCIARCODECVLPAENLAMIRADA

IAZGOLPESENLAVIDATANFVERTESZONOSE
```

Entonces, la salida proporcionada debe estar almacenada en el archivo **HERALDOSNEGROS_pre.txt** para su uso posterior.

HERALDOSNEGROS_pre.txt

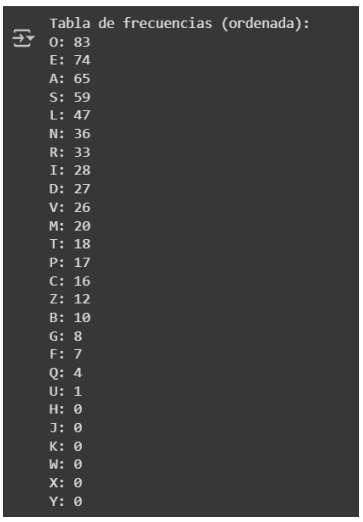
```
1  IAZGOLPESENLAVIDATANFVERTESZONOSE
2  GOLPESCOMODELODIODEDIOSCOMOSIANTEELLOS
3  LARESACADETODOLOSVFRIDO
4  SEEMPOZARAENELALMAZONOSE
5
6  SONPOCOSPEROSONABRENZANIASOSCVRAS
7  ENELROSTROMASFIEROZENELLOMOMASFVERTE
8  SERANTALVEZLOSPOTROSDEBARBAROSATILAS
9  OLOSIERALDOSNEGROSQVENOSMANDALAMVERTE
10
11 SONLASCAIDASIONDASDELOSCRISTOSDELALMA
12 DEALGVNAFEADORABLEQVEELDESTINOBLASFEMA
13 ESOSGOLPESSANGRIENTOSSONLASCREPITACIONES
14 DEALGUNPANQVEENLAPVERTADELIORNOSENOSQVEMA
15
16 ZELIOMBREPOBREPOBREVVELVELOSOIOSCOMO
17 CVANDOPORSOBREELIOMBRONOSLLAMAVNAPALMADA
18 VVELVELOSOIOSLOCOSZTODOLOVIVIDO
19 SEEMPOZACOMOCIARCODECVLPAENLAMIRADA
20
21 IAZGOLPESENLAVIDATANFVERTESZONOSE
```

5. **Abra el archivo generado e implementar una función que calcule una tabla de frecuencias para cada letra de la ‘A’ a ‘Z’. La función deberá definirse como: *frecuencias(archivo)* Deberá devolver un diccionario cuyos índices son las letras analizadas y cuyos valores son las frecuencias de las mismas en el texto (número de veces que aparecen). Reconozca en el resultado obtenido los cinco caracteres de mayor frecuencia.**

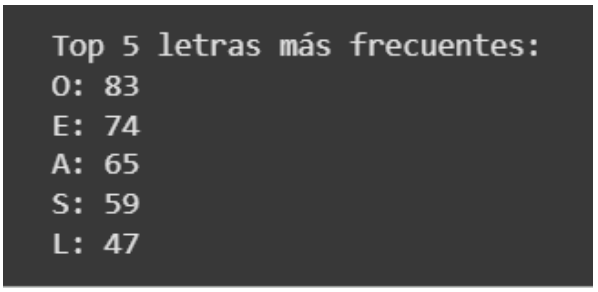
   Este nos servirá para calcular la frecuencia de cada carácter, es decir, calcular la frecuencia de aparición de cada letra del alfabeto.

```python
import string

file = "HERALDOSNEGROS_pre.txt"

def frecuencia(archivo):
  with open(archivo, "r", encoding="utf-8") as f:
    text_result = f.read().upper()

  frequency = {letter: 0 for letter in string.ascii_uppercase}
  for letter in text_result:
    if letter in frequency:
        frequency[letter] += 1
  return frequency

frequency = frecuencia(file)

print("Tabla de frecuencias (ordenada):")
for letra, freq in sorted(frequency.items(), key=lambda x: x[1],
reverse=True):
    print(f"{letra}: {freq}")

print("\nTop 5 letras más frecuentes:")
for letra, freq in sorted(frequency.items(), key=lambda x: x[1],
reverse=True)[:5]:
    print(f"{letra}: {freq}")
```

Este código va a encargarse de leer el nuevo archivo con el texto preprocesado anteriormente, luego creamos nuestra función **frecuencia** que va a devolver el diccionario con las frecuencias, luego se va a ir recorriendo cada carácter del texto, si el carácter es una letra se aumenta el contador a dicha letra para que al final devuelva cuantas apariciones tiene una letra.



Y por último mostramos los primeros 5 carácteres que tienen mayor frecuencia.



6. **Aplicar el método Kasiski, que recorre el texto preprocesado y halla los trigramas en el mismo (sucesión de tres letras seguidas que se repiten) y las distancias (número de caracteres entre ellos) entre los trigramas**

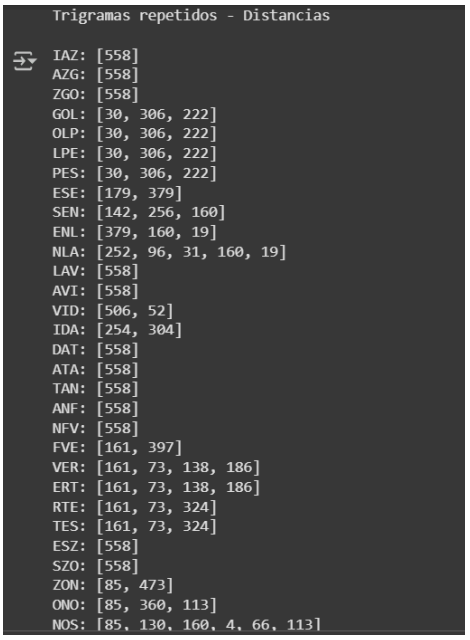
   Este método es utilizado para descifrar el cifrado Vigenère al determinar la longitud de la clave secreta. Para el ejercicio debemos recorrer el texto y hallar todos los trigramas para luego calcular las distancia entre cada trigrama encontrado.

Python

```
def kasiski_method(file):
  with open(file, "r", encoding="utf-8") as f:
    text = f.read().upper().replace("\n", "")

  trigram_positions = {}

  for i in range(len(text) - 2):
      trigram = text[i:i+3]
      if trigram in trigram_positions:
          trigram_positions[trigram].append(i)
      else:
          trigram_positions[trigram] = [i]

  repeated_trigrams = {t: pos for t, pos in trigram_positions.items() if
len(pos) > 1}

  trigram_distances = {}
  for trigram, positions in repeated_trigrams.items():
      distances = []
      for i in range(len(positions) - 1):
          distances.append(positions[i+1] - positions[i])
      trigram_distances[trigram] = distances

  return trigram_distances

result_kasiski = kasiski_method(file)

print("Trigramas repetidos - Distancias\n")
```

```
for trigram, distance in result_kasiski.items():
  print(f"{trigram}: {distance}")
```

Ahora pasamos al método **Kasiski**, primero vamos a crear nuestro trigramas que son secuencias de 3 letras seguidas, luego vamos a guardando las posiciones en que aparece el trigrama descartando los trigramas que aparecen una sola vez y conservando los que aparecen múltiples veces y calculamos la distancia entre las apariciones consecutivas de los trigramas. Acá una muestra de la salida generada.

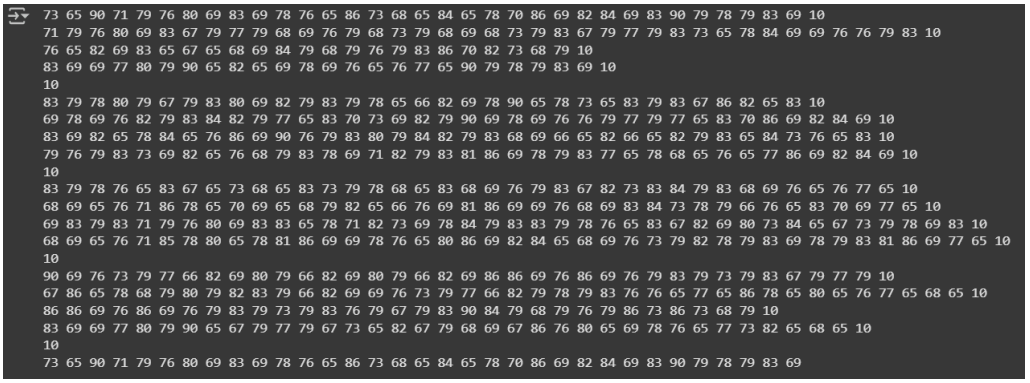


7. Volver a preprocesar el archivo cambiando cada carácter según UNICODE-8

Simplemente cambiar los carácteres a código Unicode o UTF-8.

Python
```
file_unicode8 = "HERALDOSNEGROS_unicode8.txt"

with open(file, "r", encoding="utf-8") as f:
  lines = f.readlines()

with open(file_unicode8, "w", encoding="utf-8") as f:
  for line in lines:
      unicode_values = [str(ord(c)) for c in line]
      unicode_line = " ".join(unicode_values)
      print(unicode_line)
      f.write(" ".join(unicode_values) + "\n")
```

Ahora procederemos a procesar nuestro texto con **UNICODE-8**, simplemente leemos el archivo y cada carácter que encontramos lo convertimos en su código **UNICODE** y convertirlo a texto su código. La salida que tendremos es el texto con su código.

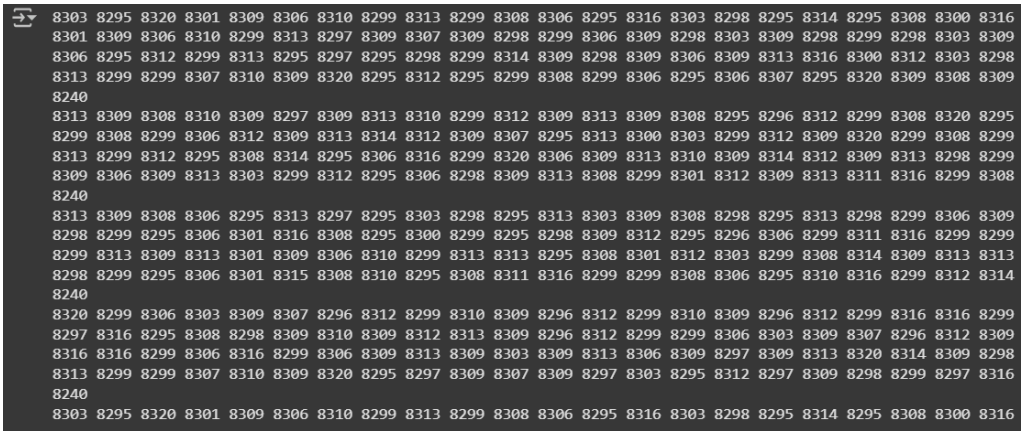


8. Volver a preprocesar el archivo cambiando cada carácter según UNICODE-8230

Similar al ejercicio anterior sería el mismo código con la diferencia de volver los carácteres a Unicode-8230.

```python
file_unicode8230 = "HERALDOSNEGROS_unicode8230.txt"

with open(file, "r", encoding="utf-8") as f:
  lines = f.readlines()

with open(file_unicode8, "w", encoding="utf-8") as f:
  for line in lines:
      unicode_values = [str(ord(c) + 8230) for c in line]
      unicode_line = " ".join(unicode_values)
      print(unicode_line)
      f.write(" ".join(unicode_values) + "\n")
```

Ahora procederemos a procesar nuestro texto con UNICODE-8230, simplemente haremos el mismo proceso que el ejercicio anterior solo que dándole un **offset** de 8239. La salida que tendremos es el texto con su código, solo una muestra debido al tamaño de la salida.



9. **Volver a preprocesar el archivo insertando la cadena AQUÍ cada 20 caracteres, el texto resultante deberá contener un número de caracteres que sea múltiplo de 4, si es necesario rellenar al final con caracteres X según se necesite.**

   Acá vamos a insertar una cadena de caracteres dividiéndolo en bloques de 20 caracteres y asegurar que la longitud de texto sea un múltiplo de 4, si no es un múltiplo de 4 se va a rellenar con el carácter X para cumplir con la regla.

```python
file_final = "HERALDOSNEGROS_aqui.txt"

with open(file, "r", encoding="utf-8") as f:
    text = f.read()

processed_text = ""
for i in range(0, len(text), 20):
    processed_text += text[i:i+20] + "AQUÍ"

remainder = len(processed_text) % 4
if remainder != 0:
    processed_text += "X" * (4 - remainder)

with open(file_final, "w", encoding="utf-8") as f:
    f.write(processed_text)

print(processed_text)
print("\nLongitud total:", len(processed_text))
```

   Finalmente insertamos la cadena **AQUÍ** cada 20 carácteres y se calcula la longitud del texto sea un múltiplo de 4, si este no es se va añadiendo los carácteres **X** que sean necesarios al final para poder ajustarlo, y también guardamos el resultado en un nuevo archivo y también mostrarlo en pantalla con su longitud total.

Cuestionario Final

1. **Describa los siguientes términos (áreas de la seguridad informática)**
    - **Protección y seguridad de los datos**

      Se refiere a estrategias y procesos de seguridad que ayudan a proteger información confidencial frente a corrupción, vulneración y pérdida. Esto se va a lograr a través de diversas estrategias y tecnologías como la encriptación de datos, el control de acceso, la autenticación multifactor y las copias de seguridad. Este es importante para mantener la organización protegida ante el robo, la filtración y la pérdida de datos. Implica tanto el uso de directivas de privacidad que satisfagan las normativas de cumplimiento como la prevención del daño a la reputación de la organización. [1]

    - **Criptografía**

      Es un proceso de ocultar o codificar información para que solo la persona a la que se dirigió un mensaje pueda leerla, las técnicas modernas incluyen algoritmos y cifrados que permiten el **cifrado** y el **descifrado** de información, como claves de cifrado de 128 bits y 256 bits. La criptografía está en la base de la sociedad moderna. Es la base de innumerables aplicaciones de Internet a través del protocolo seguro de transferencia de hipertexto (HTTPS), de la comunicación segura de texto y voz, e incluso de las monedas digitales. [2][3]

    - **Seguridad y fortificación de redes**

      Este es un término amplio para describir la protección de los recursos de cómputo ante ataques y fallas en la disponibilidad, confidencialidad e integridad. Este involucra el anti-malware, firewalls, detección de intrusiones, tecnología para la prevención de pérdida de datos y otras tecnologías. La fortificación es un proceso específico dentro de la seguridad de redes que se

enfoca en reforzar la seguridad de los dispositivos de red y servidores, como routers, switches y firewalls, para reducir su superficie de ataque y minimizar vulnerabilidades. [4]

- **Seguridad en aplicaciones informáticas, programas y bases de datos**

  La seguridad en aplicaciones informáticas es debido a las vulnerabilidades web conocidas siguen planteando riesgos y con frecuencia se vuelven a introducir en las aplicaciones durante el proceso de desarrollo de software por cada nueva generación de codificadores. A medida que las aplicaciones y las API se vuelven más complejas, crean nuevas vulnerabilidades y posibles terminales para los hackers, entonces se refiere al conjunto de prácticas, técnicas y controles orientados a proteger el software (aplicaciones y programas) así como las bases de datos de amenazas, vulnerabilidades y accesos no autorizados. [5]

  La seguridad de las bases de datos se refiere a la gama de herramientas, controles y medidas diseñados para establecer y preservar la confidencialidad, integridad y disponibilidad de las bases de datos. La confidencialidad es el elemento que se ve comprometido en la mayoría de las vulneraciones de datos. Además se dedica a proteger lo siguiente: datos en la base de datos, sistema de gestión de base de datos, cualquier aplicación asociada, servidor de base de datos físico o virtual e infraestructura informática. [6]

- **Gestión de seguridad en equipos y sistemas informáticos**

  Es el proceso de proteger la información, los sistemas y las redes contra amenazas y accesos no autorizados, mediante un conjunto de políticas, herramientas y prácticas de ciberseguridad. Esto incluye la identificación y evaluación de riesgos, la implementación de medidas como firewalls, antivirus y cifrado, el monitoreo continuo, y la respuesta ante incidentes, todo ello con el fin de asegurar la confidencialidad, integridad y disponibilidad de los activos tecnológicos. [7]

- **Informática forense**

  La informática forense es una rama de la ciberseguridad que se enfoca en la recolección, preservación, análisis y presentación de evidencia digital con el fin de utilizarla en investigaciones legales. A través de esta disciplina, los expertos forenses pueden recuperar datos eliminados, rastrear actividades sospechosas y documentar cualquier tipo de delito que ocurra en el mundo digital, este campo se ocupa de examinar cualquier dispositivo electrónico (computadoras, teléfonos móviles, discos duros, redes, etc.) para obtener evidencia que sea admisible en un tribunal. Ya sea que se trate de un fraude financiero, robo de identidad o ciberataque, la informática forense puede

brindar respuestas y, lo más importante, pruebas sólidas para llegar a la verdad. [8]

- **Ciberdelito, ciberseguridad**

  El **ciberdelito** es una actividad delictiva que se dirige o utiliza una computadora, una red informática o un dispositivo conectado. La mayoría de los ciberdelitos son cometidos por ciberdelincuentes o hackers que buscan lucrarse. Sin embargo, en ocasiones, el ciberdelito busca dañar computadoras o redes por motivos ajenos al lucro, como pueden ser políticos o personales. [9]

  Mientras que la **ciberseguridad** es la práctica de proteger equipos, redes, aplicaciones de software, sistemas críticos y datos de posibles amenazas digitales. Las organizaciones tienen la responsabilidad de proteger los datos para mantener la confianza del cliente y cumplir la normativa. Utilizan medidas y herramientas de ciberseguridad para proteger los datos confidenciales del acceso no autorizado, así como para evitar interrupciones en las operaciones empresariales debido a una actividad de red no deseada. [10]

2. **Describa los siguientes términos (áreas de la seguridad de la información)**
   - **Gestión de la seguridad de la información**

     Se refiere a un conjunto de procesos, políticas, procedimientos y medidas que adopta una organización para la protección de la información que posee la misma organización, ya sea en formato físico o digital. El objetivo de la GSI es garantizar la confidencialidad, integridad y disponibilidad de la información, así como protegerla contra amenazas internas y externas. [11]

   - **Asesoría y auditoría de la seguridad**

     Una auditoría de seguridad informática es un procedimiento que evalúa el nivel de seguridad de una empresa o entidad, analizando sus procesos y comprobando si sus políticas de seguridad se cumplen. El principal objetivo de una auditoría de seguridad es detectar las vulnerabilidades y debilidades de seguridad que pueden ser utilizadas por terceros malintencionados para robar información, impedir el funcionamiento de sistemas, o en general, causar daños a la empresa. [12]

   - **Análisis y gestión de riesgos**

     Esta área consiste en identificar, evaluar, priorizar los riesgos que pueden afectar los activos de información, decidir qué hacer con esos riesgos (aceptarlos, mitigarlos, transferirlos, evitarlos) y monitorear los controles implementados. En los últimos meses, ha aumentado el interés de las organizaciones peruanas en la adecuación a la Ley de Protección de Datos

Personales (LPDP), establecida por la Ley Nº 29733. Esta normativa es de cumplimiento obligatorio desde el 8 de mayo de 2015, fecha límite establecida para su adecuación. Sin embargo, muchas organizaciones enfrentan dificultades para cumplir con la ley, exponiéndose a sanciones. Recientemente, se han abierto Procedimientos Administrativos Sancionadores contra tres organizaciones, según información publicada por el Ministerio de Justicia y Derechos Humanos. [13]

- **Continuidad de negocio**

  Se refiere a los planes, políticas y procedimientos que permiten a una organización mantener o restablecer sus funciones críticas después de un incidente, desastre o interrupción. Este tiene los siguientes componentes:

  - Identificación de los procesos críticos de negocio.
  - Análisis de impacto al negocio (BIA – Business Impact Analysis): cuánto daño temporal o económico generaría la interrupción de esas funciones.
  - Desarrollo de estrategias de recuperación: respaldo, redundancias, sitios alternativos, recuperación de datos.
  - Planes de contingencia y recuperación ante desastres.
  - Pruebas y simulacros con periodicidad para verificar que los planes funcionan. [14]

- **Buen gobierno**

  Es la implementación de protocolos y tecnologías seguras, también es el marco de políticas, procesos y estrategias que una organización implementa para gestionar y proteger sus datos y activos de información de manera eficaz y alineada con sus objetivos de negocio, asegurando la rendición de cuentas, la transparencia y la capacidad de respuesta ante amenazas y cambios constantes en el entorno digital. [15]

- **Comercio electrónico**

  Son las medidas y protocolos de protección que las empresas en línea tienen para proteger y salvaguardar las transacciones en línea, información de los clientes y todas las operaciones comerciales de ciberataques, violaciones de datos y acceso no autorizado. Al dar prioridad a la seguridad en el comercio electrónico, las empresas pueden generar confianza entre sus clientes, garantizar la integridad de sus transacciones y mantener un entorno de compra en línea seguro. [16]

- **Legislación relacionada con seguridad**

  Es el conjunto de medidas preventivas y reactivas para cuidar la información personal o de valor de una persona u organización y evitar que caiga en manos de extraños y sea usada ilícitamente.

La legislación clave relacionada con la seguridad de la información en Perú incluye la Ley N° 29733 de Protección de Datos Personales y su reglamento para proteger la información personal de los ciudadanos, la Ley N° 30171 de Delitos Informáticos que sanciona las acciones ilícitas contra sistemas de información, la Ley de Gobierno Digital (D.L. 1412) y sus reglamentos asociados para la gestión de la seguridad digital, así como la normativa de ciberseguridad. [17]

3. **Describa alguna otra operación o función de preprocesamiento que se implemente sobre el texto claro en los criptosistemas, en que afecta la complejidad de estas funciones al desempeño del mismo.**

   Una operación adicional de preprocesamiento común en criptosistemas es **la permuta o transposición del texto claro**. Esta técnica consiste en reordenar los caracteres del mensaje original según un patrón o algoritmo específico, como agrupar caracteres en bloques y cambiar el orden dentro de cada bloque. Por ejemplo, el texto se puede dividir en bloques de 5 caracteres y luego cambiar el orden de los caracteres en cada bloque con una permutación fija, como 43521, donde el cuarto carácter de cada bloque se coloca primero, luego el tercero, el quinto, el segundo y, por último, el primero.

   Esta operación puede afectar la complejidad de un criptosistema ya que al modificar la estructura interna preservando los carácteres originales pero alterando su posición dificulta el análisis y ataque basado en frecuencia o patrones comunes del texto. También, su implementación agrega una capa extra de procesamiento aumentando el tiempo de cifrado y descifrado que depende de la longitud del texto y complejidad. [18][19]

4. **Describa la máquina enigma, luego muestre usando un simulador en Internet la encriptación de la frase QUERIDA HIJA, para tres posiciones distintas de los rotores.**

   Esta máquina fue diseñada para crear mensajes cifrados complejos que eran casi imposible de descifrar. Antes y durante la Segunda Guerra Mundial, los alemanes junto a sus aliados usaban estas máquinas Enigma para el envío de mensajes militares. Su diseño permitía millones de combinaciones posibles para codificar mensajes, lo que la hacía parecer invulnerable a los intentos de descifrado. Esta máquina con cada pulsación hace avanzar al menos el rotor derecho, lo que modifica la sustitución usada para la tecla siguiente; además hay un reflector que devuelve la señal por los rotores, lo que hace que la máquina sea autor recíproca (el mismo ajuste cifra y descifra). La Enigma normalmente incorporaba también un plugboard (steckerbrett) que intercambiaba pares de letras antes y después del conjunto de rotores, añadiendo muchísima más complejidad al espacio de claves

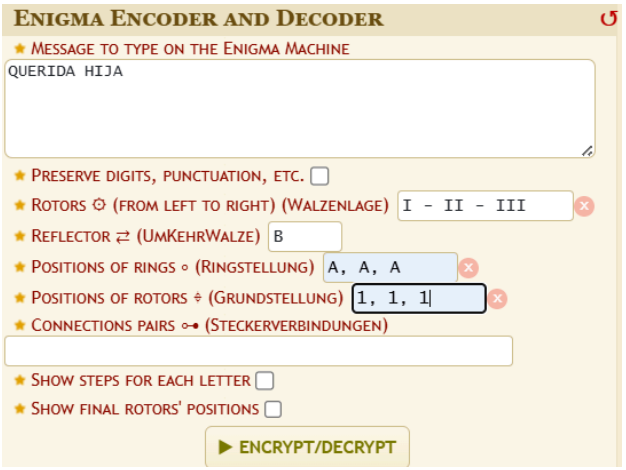
   Sin embargo, el aparente “indescifrable” código Enigma tuvo una vulnerabilidad: su dependencia de patrones repetitivos y configuraciones iniciales, ya que matemáticos

de Polonia, Francia y el Reino Unido desarrollaron técnicas avanzadas que descifraron el código Enigma, ayudando a acortar la guerra. [20][21]

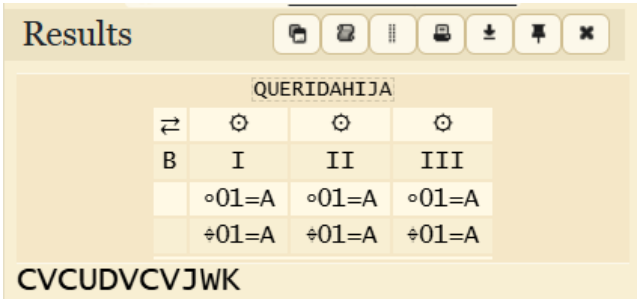
Ahora llevemos un ejemplo en el simulador **dCode** que es para simular Enigma probando diferentes posiciones de los rotores.

a) Posición A, A, A

Probaremos el simulador con los rotores en la posición AAA.

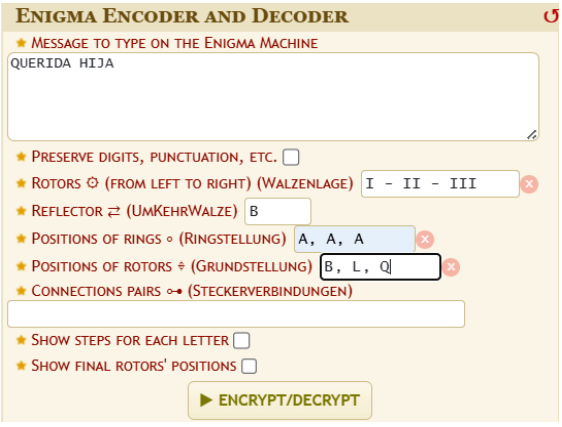


Entonces al ejecutar el encriptado nos sale el siguiente mensaje.

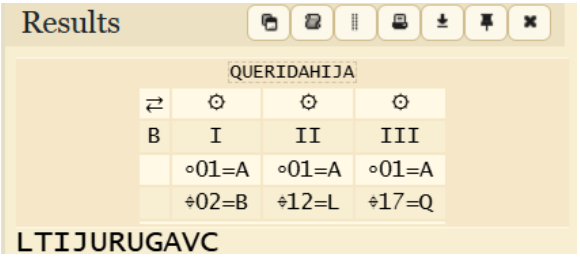


b) Posición B, L, Q

Probaremos el simulador con los rotores en la posición BLQ.

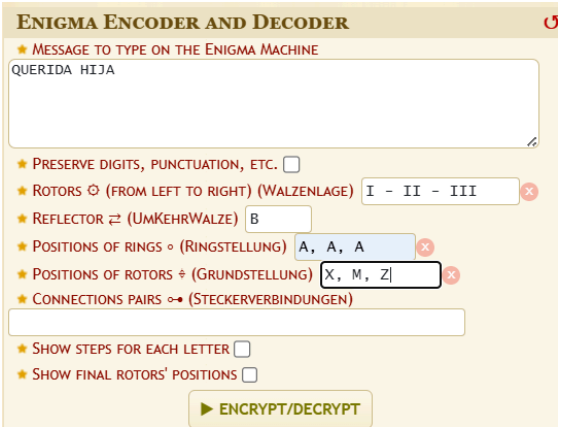


Entonces al ejecutar el encriptado nos sale el siguiente mensaje.

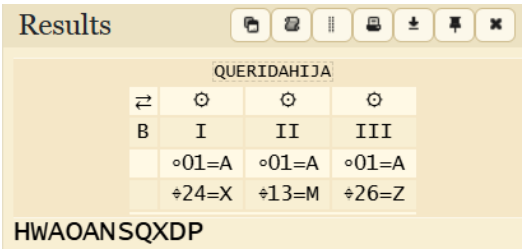


c) Posición X, M, Z

Probaremos el simulador con los rotores en la posición BLQ.



Entonces al ejecutar el encriptado nos sale el siguiente mensaje.



5. Describa la aplicación de Unicode-8

UTF-8 es un sistema de codificación para Unicode, este significa **Formato de Transformación Unicode - 8 bit** que puede traducir cualquier carácter Unicode a una cadena binaria única coincidente, y también puede convertir la cadena binaria de vuelta a un carácter Unicode. Este usa una longitud variable de bytes: de 1 a 4 bytes por carácter, dependiendo del punto de código (code point) de Unicode y es compatible con ASCII: los primeros 128 caracteres Unicode (U+0000 hasta U+007F) se representan exactamente igual que en ASCII, con un solo byte. [22]

Este es utilizado en las siguientes aplicaciones:

- Páginas web y navegación.

- Protocolos de comunicación.
- Bases de Datos.
- Sistemas Operativos y Software.
- Localización.

Conclusiones

- Consiste en limpiar y preparar los datos textuales para obtener una representación computacional más rica, menos ambigua y semánticamente significativa. [23]
- Incluye tareas como la tokenización, normalización, eliminación de palabras irrelevantes, stemming, lematización y eliminación de ruido. [23]
- Este mejora significativamente la precisión de tareas de clasificación automática de textos y otros modelos de aprendizaje automático. [24]
- La normalización de texto ayuda a tratar diferentes formas de una misma palabra como equivalentes, facilitando la agrupación y análisis semántico. [24]
- El preprocesamiento es clave para reducir el ruido y la complejidad en datos textuales informales, o con errores, mejorando la interpretación y automatización en sistemas. [25]
- Al ocultar patrones comunes del lenguaje, se reduce la eficacia de ataques basados en frecuencia o diccionarios. [25]
- La normalización del texto claro asegura que diferentes plataformas y algoritmos trabajen sobre un mismo formato, evitando errores por diferencias de codificación o alfabeto. [24]
- Es esencial para garantizar que los datos textuales sean de alta calidad, estén bien estructurados y listos para ser procesados eficientemente por modelos de machine learning y herramientas de análisis de texto. [24]

Bibliografía

[1] “¿Qué es la protección de datos? | Seguridad de Microsoft”. Your request has been blocked. This could be due to several reasons. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://www.microsoft.com/es-es/security/business/security-101/what-is-data-protection

[2] “¿Qué es la criptografía? Definición, importancia, tipos | Fortinet”. Fortinet. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://www.fortinet.com/lat/resources/cyberglossary/what-is-cryptography

[3] “¿Qué es la criptografía? - Explicación sobre la criptografía - AWS”. Amazon Web Services, Inc. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://aws.amazon.com/es/what-is/cryptography/

[4] “¿Qué es la seguridad de redes?” Trend Micro. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://www.trendmicro.com/es_mx/what-is/network-security.html

[5] “¿Qué es la seguridad de aplicaciones? | Akamai”. Akamai. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://www.akamai.com/es/glossary/what-is-app-security

[6] IBM. “Seguridad de bases de datos: Una guía esencial | IBM”. IBM. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://www.ibm.com/es-es/think/topics/database-security

[7] E. Martín. “¿Qué es la seguridad informática y cómo implementarla?” Grupo Cibernos. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://www.grupocibernos.com/blog/que-es-la-seguridad-informatica-y-como-implementarla

[8] C. Cañon. “Informática forense: ¿qué es y por qué es importante? | universidad san marcos”. Universidad San Marcos. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://www.usanmarcos.ac.cr/blogs/informatica-forense-que-es-y-por-que-es-importante

[9] “What is cybercrime and how to protect yourself?” Kaspersky. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://www.kaspersky.com/resource-center/threats/what-is-cybercrime

[10] “Ciberseguridad ante el crimen transnacional y la ciberdelincuencia [parte 1/4]”. LISA Institute. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://www.lisainstitute.com/blogs/blog/ciberseguridad-crimen-transnacional-ciberdelincuencia

[11] “Gestión de la seguridad de la información GRC”. GRCTools. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://grctools.software/soluciones/ciberseguridad/gestion-de-la-seguridad-de-la-informacion/

[12] A. I. Team. “¿Qué es una auditoría de seguridad informática? Tipos y Fases”. Ambit Iberia | Consultoría regulatoria y de calidad en sector salud. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://www.ambit-iberia.com/blog/qué-es-una-auditoría-de-seguridad-informática-tipos-y-fases

[13] “El análisis de riesgos en la ley de protección de datos personales en perú”. GlobalSuite Solutions. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://www.globalsuitesolutions.com/es/el-analisis-de-riesgos-en-la-ley-de-proteccion-de-datos-personales-en-peru/

[14] “Plan contingencia continuidad negocio”. Incibe. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://www.incibe.es/empresas/que-te-interesa/plan-contingencia-continuidad-negocio

[15] “¿Qué es la gobernanza de seguridad de la información en ciberseguridad?” Kiteworks | Your Private Data Network. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://www.kiteworks.com/es/gestion-de-riesgos-de-ciberseguridad/gobernanza-seguridad-en-ciberseguridad/

[16] “Seguridad de los datos del comercio electrÃ³nico: PrevenciÃ³n de la ciberdelincuencia”. Alumio | Integration Platform for Commerce Connectivity. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://www.alumio.com/es/blog/e-commerce-security-prioritize-data-security-and-prevent-cyber-attacks

[17] “Que es la seguridad de la información”. Gobierno del Perú. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://www.gob.pe/23391-que-es-la-seguridad-de-la-informacion

[18] “Criptografía en la seguridad de sistemas informáticos”. Dspace de la Universidad del Azuay: Página de inicio. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://dspace.uazuay.edu.ec/bitstream/datos/2142/1/04282.pdf

[19] “Algoritmos de cifrado moderno”. TIC. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://imaster.academy/contenidos-tematicos/talentotech/TalentoTech/M2unidad1/BlockChain/Explorador/assets/files/Leccin-2.Misin2Algoritmosdecifradomoderno.pdf

[20] “The enigma machine — the national museum of computing”. The National Museum of Computing. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://www.tnmoc.org/bh-2-the-enigma-machine

[21] “Alan turing: La maquina enigma y su legado en la ciberseguridad”. Devel Group. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://devel.group/blog/alan-turing-la-maquina-enigma-y-su-legado-en-la-ciberseguridad/

[22] “What is UTF-8 Encoding? A Guide for Non-Programmers”. HubSpot Blog | Marketing, Sales, Agency, and Customer Success Content. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://blog.hubspot.com/website/what-is-utf-8

[23] “Pre-Procesamiento y Representación de texto”. UNSD - Welcome to UNSD. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://unstats.un.org/capacity-development/data-for-now/training-materials/preprocess_text_representation.pdf

[24] “¿Qué es el preprocesamiento de datos para el aprendizaje automático? | Pure Storage”. The Data Platform | Pure Storage. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://www.purestorage.com/la/knowledge/what-is-data-preprocessing.html

[25] “Preprocesamiento de datos: Conceptos, importancia y herramientas | Astera”. Astera. Accedido el 21 de septiembre de 2025. [En línea]. Disponible: https://www.astera.com/es/type/blog/data-preprocessing/