Universidad Central del Ecuador

FICA

Criptografía y Seguridad

Nombres:

Andrango Alex

Mullo Bryan

Paredes Juan Carlos

Quisilema Anddy

Ramos John Felipe

Tema:

Algoritmos Criptográficos - Documentación

Fecha:

30/04/2024

Algoritmo 1: Algoritmo que escriba todas las permutaciones posibles de una palabra de longitud *n* SIN espacios (Anagrama). La palabra se ingresa al iniciar el algoritmo. El algoritmo debe mostrar el número total de permutaciones y las 10 primeras ordenadas alfabéticamente.

Descripción del Problema
El problema consiste en encontrar todas las permutaciones posibles de una palabra dada sin espacios, es decir, generar todos los anagramas de la palabra. Se debe mostrar el número total de permutaciones y las primeras 10 permutaciones ordenadas alfabéticamente.

Análisis del Algoritmo
El algoritmo implementado utiliza la generación de permutaciones basado en intercambios y reversión. Aquí se describe brevemente el funcionamiento del algoritmo:

1. Entrada de Datos: El usuario ingresa una palabra desde la consola.

2. Ordenamiento de Caracteres: Los caracteres de la palabra se ordenan alfabéticamente para garantizar que las permutaciones estén en orden lexicográfico.

3. Generación de Permutaciones: Se utiliza un bucle do-while para generar las permutaciones

una por una. Dentro del bucle, se incrementa un contador para llevar el registro del número total de permutaciones.

4. Mostrar las Primeras 10 Permutaciones: Durante la generación de permutaciones, se muestran las primeras 10 permutaciones ordenadas alfabéticamente.

5. Mostrar el Número Total de Permutaciones: Una vez que se generan todas las permutaciones, se muestra el número total de permutaciones.

Explicación del Código
El código se realizó utilizando el lenguaje de programación Java. A continuación, se muestran explicaciones breves de las partes clave del código:

• Entrada de Datos: Utiliza un objeto Scanner para leer la palabra ingresada por el usuario desde la consola.

• Ordenamiento de Caracteres: La palabra se convierte en un arreglo de caracteres y se ordena alfabéticamente utilizando el método Arrays.sort.

• Generación de Permutaciones: Se utiliza el método realizado con anterioridad para generar las permutaciones de forma iterativa.

• Mostrar las Permutaciones: Se muestran las primeras 10 permutaciones dentro del bucle do-while.

• Mostrar el Número Total de Permutaciones: Al final, se muestra el número total de permutaciones generadas.

Conclusión
El algoritmo implementado es capaz de encontrar todas las permutaciones posibles de una palabra dada y mostrar las primeras 10 permutaciones ordenadas alfabéticamente. El enfoque basado en intercambios y reversión permite generar las permutaciones de manera eficiente

Algoritmo 2: Algoritmo que realice el cifrado de un mensaje por permutación de filas, teniendo como clave n filas. Tanto n como el texto del mensaje se ingresan al iniciar el algoritmo. El algoritmo debe controlar que el número de caracteres del mensaje (sin espacios), sea menor o igual que n x n. Imprima la matriz de cifrado, el mensaje original y el mensaje cifrado. Si en la matriz de cifrado sobran espacios para almacenar los caracteres del mensaje original, estos deben llenarse con "*".

Este código implementa un algoritmo de cifrado básico que utiliza una matriz cuadrada para reorganizar un mensaje de entrada de acuerdo con el número de filas especificado.

• Se solicita al usuario que ingrese un mensaje y se elimina cualquier espacio en blanco del mensaje usando replaceAll("\\s+", "").

• Se solicita al usuario que ingrese el número de filas (n) para la matriz de cifrado.

• Se calcula el número total de caracteres en el mensaje (caracteresMensaje) y el número total de caracteres en la matriz de cifrado (caracteresMatriz).

• Se comprueba si el mensaje cabe en la matriz de cifrado. Si no cabe, se imprime un mensaje y se termina el programa.

• Se crea una matriz de caracteres (matrizCifrado) con dimensiones n x n para almacenar el mensaje cifrado.

• Se inicializa un índice (indiceMensaje) para rastrear la posición actual en el mensaje.

• Se llena la matriz de cifrado con los caracteres del mensaje. Si el mensaje es más corto que el tamaño de la matriz, se llenan los espacios restantes con '*'.

• Se imprime la matriz de cifrado en la consola.

• Se imprime el mensaje original.

• Se cifra el mensaje reorganizando los caracteres de la matriz de cifrado en una cadena (mensajeCifrado).

• Se imprime el mensaje cifrado en la consola.

Dentro de las técnicas mencionadas a lo largo del artículo, las más utilizadas para implementaciones eficientes son las variantes de las representaciones con signo en un contexto general.

Si se desea aplicar sobre un conjunto de curvas específicas, existen métodos que aprovechan las características de dichas curvas optimizando los tiempos de respuesta de los algoritmos. En las curvas definidas sobre campos binarios, métodos como el de reducción a la mitad o Point Halving, ofrecen tiempos de respuesta muy reducidos. [1]

Algoritmo 3: Algoritmo que realice el cifrado de un mensaje por permutación de columnas, teniendo como clave n columnas. Tanto n como el texto del mensaje se ingresan al iniciar el algoritmo. El algoritmo debe controlar que el número de caracteres del mensaje (sin espacios), sea menor o igual que n x n. Imprima la matriz de cifrado, el mensaje original y el mensaje cifrado. Si en la matriz de cifrado sobran espacios para almacenar los caracteres del mensaje original, estos deben llenarse con "*".

Existen infinidad de algoritmos criptográficos que, partiendo de un documento original, obtienen otro documento o conjunto de información. Los algoritmos más conocidos son los que obtienen un documento a partir de un documento original al aplicar un algoritmo que utiliza una clave secreta como argumento. En general los algoritmos criptográficos se pueden clasificar en tres grandes familias [2].

- Criptografía de clave secreta o criptografía simétrica.
- Criptografía de clave pública o criptografía asimétrica.
- Algoritmos HASH o de resumen.

Para la resolución de este ejercicio se utiliza el enfoque de la criptografía que es el cifrado de transposición.

Este tipo de cifradores eran muy usados en la criptografía clásica y por tanto, al tener que hacer los cálculos por medios muy básicos, normalmente el algoritmo se basaba en un diseño geométrico o en el uso de artilugios mecánicos (Ej escítala). Este tipo de algoritmos son de clave simétrica porque es necesario que tanto el que cifra como el que descifra sepan la misma clave para realizar su función. La clave puede ser intrínseca en el propio método de cifrado/descifrado de forma que algoritmo y clave son un conjunto indivisible.

Un cifrado de transposición es aquel en el que se cambia el orden de los caracteres para oscurecer el mensaje [2].

Una versión temprana de un cifrado de transposición era un Scytale, en el que se envolvía papel alrededor de un palo y se escribía el mensaje. Una vez desenvuelto, el

mensaje sería ilegible hasta que el mensaje se envolviera de nuevo alrededor de un palo del mismo tamaño.

Un cifrado de transposición moderno se realiza escribiendo el mensaje en filas, luego formando el mensaje cifrado a partir del texto en las columnas [3].

Algoritmo 4: Algoritmo que realice el cifrado de una cadena de caracteres mediante un método de sustitución Monoalfabético de desplazamiento n caracteres a la derecha. Tanto la palabra como el valor de n se ingresan al iniciar el algoritmo. El algoritmo debe mostrar el alfabeto original, el alfabeto cifrado, la cadena de caracteres ingresada y su resultado.

¿Qué es el cifrado César y cómo funciona?

Un cifrado César es una de las técnicas de cifrado más simples y conocidas.

Lleva el nombre de Julio César, es uno de los tipos de cifrados más antiguos y se basa en el cifrado monoalfabético más simple. Se considera un método débil de criptografía, ya que es fácil decodificar el mensaje debido a sus técnicas de seguridad mínimas.

Por la misma razón, un cifrado César a menudo se incorpora solo en partes de otros esquemas de cifrados complejos.

En criptografía, un cifrado César se clasifica como un cifrado por sustitución en el que el alfabeto en el texto plano se desplaza por un número fijo en el alfabeto. [5].

Algoritmo 5: Algoritmo que realice el cifrado de una cadena de caracteres mediante un método de sustitución Polialfabético de Vigenère. La cadena se ingresa al iniciar el algoritmo. El algoritmo debe mostrar la cadena de caracteres ingresada, la clave de cifrado y la cadena de caracteres cifrada.

El cifrado Vigenère es un algoritmo criptográfico estándar. Este algoritmo, bastante simple de utilizar, emplea la sustitución, similar al cifrado César, para encriptar el texto del mensaje. Una desventaja del cifrado Vigenère extendido es que puede ser determinado utilizando un método conocido como el método Kasiski. En el cifrado Vigenère, ciertas frases se repiten en el texto cifrado generado a partir del proceso de encriptación, lo que permite al método Kasiski determinar el texto original. Los códigos Goldbach, por otro lado, son un algoritmo de compresión que se emplea para abordar las debilidades del cifrado Vigenère.[6]

Algoritmo 6: Algoritmo que realice el cifrado de una cadena de caracteres utilizando la siguiente tabla de cifrado:

*	A	S	D	F	G
Q	a	b	c	d	e
W	f	g	h	i	j
E	k	l	m	n	o
R	p	q	r	s	t
T	u	v	x	y	z

La cadena de caracteres se ingresa al iniciar el programa. Si algún caracter del texto no existe en la matriz, coloque "**". Imprima la matriz de cifrado, el mensaje original y el mensaje cifrado.

Tiene un estilo de cifrado similar al de polialfabético o de Vigenere.

Bibliografía:

[1] G. G. Paredes, "Introducción a La Criptografía," Revista Digital Universitaria, vol. 7, 2006, doi: 10.1017/CBO9781107415324.004.

[2] «BALANCED SCORECARD The balanced scorecard translating strategy into action [1996].pdf». Accedido: 20 de noviembre de 2023. [En línea]. Disponible en: http://www.untag-smd.ac.id/files/Perpustakaan_Digital_1/BALANCED%20SCORECARD%20The%20balanced%20scorecard%20translating%20strategy%20into%20action%20%5B1996%5D.pdf

[3] «16.3: Cifrados de transposición», LibreTexts Español. Accedido: 30 de abril de 2024. [En línea]. Disponible en: https://espanol.libretexts.org/Matematicas/Matematicas_Aplicadas/Las_matematicas_en_la_sociedad_(Lippman)/16%3A_Criptograf%C3%ADa/16.03%3A_Cifrados_de_transposici%C3%B3n

[4] «Cifrado por transposición», Wikipedia, la enciclopedia libre. 27 de marzo de 2024. Accedido: 30 de abril de 2024. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=Cifrado_por_transposici%C3%B3n&oldid=159057636

[5] A. González. "¿Qué es el cifrado César y cómo funciona?" Ayuda Ley Protección Datos. Accedido el 30 de abril de 2024. [En línea]. Disponible: https://ayudaleyprotecciondatos.es/2020/06/10/cifrado-cesar/

[6]J. Smith et al., "Placeholder Text: A Study", Citation Styles, vol. 3, jul. 2021, doi: 10.10/X.