

MANUAL TECNICO

PRACTICA 1

TABLA DE CONTENIDO

0. Objetivo.....	3
0.1 Definiciones.....	3
Metodos	4 – 5
Librerias	6

INTRODUCCION

0. Objetivo:

Otorgar soporte al usuario que comprenda y entienda cada uno de los métodos creados para el funcionamiento del programa.

0.1 Definiciones:

PRACTICA 1: es un programa que permite la interacción del usuario con la criptografía que es una técnica que involucra diseño de métodos que protegen documentos y datos, utilizando operaciones entre matrices.

METODOS

El presente manual esta creado con fines de que el usuario reconozca cada método utilizado en la creación del programa como así mismo saber cuál es su función(requisito obligatorio tener instalado java en el equipo para poder ejecutarlo).

- Método "RAtaque"

Método que emplea las funciones FileWriter y PrintWriter para la creación y escritura dentro del archivo, a través de estas dos funciones se realizó la escritura del reporte ataque al texto plano.

- Método "RDesencriptado"

Método que emplea las funciones FileWriter y PrintWriter para la creación y escritura dentro del archivo, a través de estas dos funciones se realizó la escritura del reporte desencriptado.

- Método "REncriptado"

Método que emplea las funciones FileWriter y PrintWriter para la creación y escritura dentro del archivo, a través de estas dos funciones se realizo la escritura del reporte encriptado.

- Método "Reportes"

Método que emplea un condicional swtich para preguntar al usuario si todos los datos que se le solicitan fueron llenados de ser su respuesta si se generan los reportes solicitados y si su respuesta es no se le pide que ingrese los datos necesarios para generar los reportes.

- Método "numerosaletras"

Método que almacena en un switch con toda la conversión de los números a letras según el valor correspondiente para cada una.

- Método "conversionletrasanumeros"

Método que almacena en un switch con toda la conversión de las letras con su valor correspondiente en números según el mensaje ingresado.

- Método "codificar"

Método que emplea una secuencia de operadores que permite leer el mensaje ingresado por el usuario separarlo por caracteres almacenándolos en un vector para llamar al método "conversionletrasanumeros" para que según sea

la letra almacenada nos imprima el número correspondiente (para el espacio se utilizó el default del switch ingresado) y finalmente guardarlos en la matriz M la matriz mensaje.

- Método "MA"
Método que emplea las librerías "java.io.BufferedReader, java.io.File, java.io.FileReader" para poder realizar la lectura de un archivo de extensión "txt" que el usuario ingresa con el fin de almacenar en una matriz de dimensiones 3x3 los números de dicho que dicho documento contiene, este método utiliza un .split(",") para saber cuando es un valor diferente.
- Método "MB"
Método que emplea las librerías "java.io.BufferedReader, java.io.File, java.io.FileReader" para poder realizar la lectura de un archivo de extensión "txt" que el usuario ingresa con el fin de almacenar en una matriz de dimensiones 3xn (para encontrar el número de columnas utiliza un contador y una función .length) los números de dicho que dicho documento contiene, este método utiliza un .split(",") para saber cuándo es un valor diferente .
- Método "ENCRYPTAR"
Método que emplea la operación de $(A * M + B = C)$ para obtener la matriz c que es el mensaje cifrado, para obtener dicha matriz se utilizaron operaciones entre matrices como la multiplicación donde "ncolumnas de la matriz A deben ser igual a nfilas de la matriz B o viceversa" y la suma de matrices "ambas matrices deben tener las mismas dimensiones".
- Método "DESENCRIPTAR"
Método que realiza la operación contraria del método ENCRYPTAR este método utiliza la operatoria de $(A^{-1} * C - B = M)$ para obtener la matriz M o bien la matriz mensaje, el proceso de la obtención de la matriz inversa se utilizó Gauss-Jordán y operaciones entre matrices como la multiplicación de matrices y la resta de matrices.
- Método "inversa"
Método utilizado para llamar al método DESENCRIPTAR utilizando la matriz "A".
- Método "main"
Método que contiene toda la interfaz del menú y es lo primero que ejecuta el programa al iniciar.

LIBRERIAS

- java.io. BufferedReader

Es clase de Java para leer el texto de una secuencia de entrada (como un archivo) almacenando en el búfer caracteres que leen a la perfección caracteres, matrices o líneas.

- java.io. File

Es una clase de java que nos permite crear, eliminar archivos, directorios o bien para obtener información sobre archivos y directorios.

- java.io.FileReader

Es una clase de java que nos permiten leer y escribir "streams" de datos y de archivos bien para abrir un fichero de texto para leer usando la clase FileReader.

- java.io.FileWriter

Esta clase es usada para escribir caracteres en archivos. Su método write() permite escribir caracteres o strings a un fichero.

- java.io.PrintWriter

Es una clase que sirve para crear y escribir archivos en Java que soporta diferentes tipos de datos para imprimir desde los más primitivos como números, texto hasta arrays y objetos.

- java.util.InputMismatchException

Es una excepción lanzada por la clase Scanner cuando el elemento recibido no corresponde al tipo de dato esperado.

- java.util.Scanner

es una clase utilizada para obtener la entrada de los tipos como int, double, string etc.

- java.time.ZonedDateTime

es una ³⁴clase que nos permite representar una fecha y hora con una zona horaria.

- java.time.format.DateTimeFormatter

Este se utiliza para definir el tipo de formato que se maneja (dd/MMM/yy)