**INTERPRETACIÓN DEL DESAFÍO**

Para este desafío se nos ha pedido brindar un programa capaz de generar una cerradura la cual denotaremos como X que estará compuesta por varias estructuras de datos alineadas entre sí, que pueda ser abierta por una llave que denotaremos como k la cual es proporcionada por el usuario.

Las características que solicitan que X posea son las siguientes:

* Es una de las salidas del programa.
* Es una estructura de datos que contine dentro de sí varias estructuras de datos bidimensionales llamadas M.
* Todas las M deben estar alineadas entre sí.
* M debe poseer igual número de columnas y filas.
* El número de columnas y filas debe ser un numero impar.
* No existe un límite para el número de estructuras M.
* No existe un límite para las dimensiones de M.
* M tienen cuatro estados posibles, para que una matriz pase de un estado al siguiente se bebe de rotar noventa grados a la izquierda, dichos estados son: neutro, 1, 2, 3.
* M se ven como una matriz cuadrada de tamaño nxn.
* El contenido de M es los numero de 1 hasta nxn – 1, a excepción de la celda del medio la cual permite alinear todas las estructuras, por lo tanto, queda a nuestro criterio decidir que elemento especial irá ahí.
* La manera de rellenado de M es empezando desde la primera posición y desplazándonos hacia la derecha, una vez se llega a la última columna el siguiente número irá en la primera columna, pero en la siguiente fila, nuevamente se aclara que en el centro no se debe poner el numero siguiente a la celda anterior sino pasar hasta la celda después del centro.

Las características que solicitan que k posea son las siguientes:

* Es una entrada del programa.
* El número de elementos de k depende del número de estructuras M que es N.
* El número de elementos de k es igual a N+1.
* Los dos primeros elementos de k corresponden a las coordenadas en la primera M de X, donde el primer elemento es la fila y el segundo es la columna.
* Los demás elementos de k son las comparaciones que se deben hacer entre los elementos de los arreglos M que están alineados con la coordenada de la primera estructura M suministrada en los dos primeros elementos de k.
* Las posibles comparaciones se realizan con base en la estructura M correspondiente y la que se alinea detrás de ella M’, hay tres comparaciones validas y son: 1 para comparar si el elemento de la matriz M es mayor que el elemento de M’; 0 para comparar si el elemento de la matriz M es igual que el elemento de M’; y -1 para comparar si el elemento de la matriz M es menor que el elemento de M’.

Las salidas solicitadas para el programa además de la cerradura X son las dimensiones de cada estructura M que compone X y también las rotaciones que se deben aplicar a cada estructura para que se cumplan las condiciones de la llave.

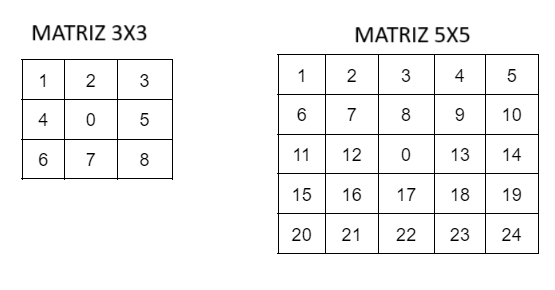
**DESGLOCE DE LA SOLUCIÓN PROPUESTA**

Como primero se empezó analizando como es el proceso de alineación entre estructuras M, de la información suministrada sabemos que se alinean varias M una tras otra, usando la celda del centro como referencia. Además de cómo se mencionó antes no existe restricción para su cantidad y sus tamaños pueden ser diferentes. Por ejemplo, puede haber una cerradura X (5, 7, 5, 9) con cuatro estructuras alineadas, de tamaño 5x5, 7x7, 5x5 y 9x9, respectivamente

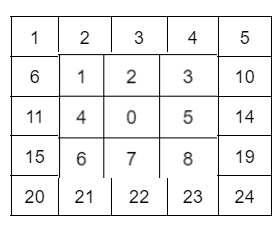
El razonamiento que se hizo acerca del alineamiento de cada celda se miró de la siguiente forma.

Para simplificar el proceso, dicho alineamiento se hará por pares de matrices, de tal forma que para los siguientes pasos (de comparación) serán más sencillos de verificar.

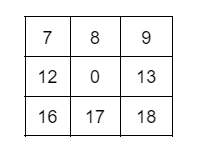
El proceso más importante es entender que es tener dos matrices alineadas por un punto de referencia, para poder ilustrar lo que pensamos se decidió hacer un dibujo:



Tenemos dos matrices, una de 3x3 y 5x5. Y la forma en las que vamos a alinearlas es sobreponer una en la otra para que así solo tengamos en cuenta los valores que necesitamos:



Así nuestra matriz de 5x5 se va a convertir en una matriz 3x3, pero manteniendo sus valores originales

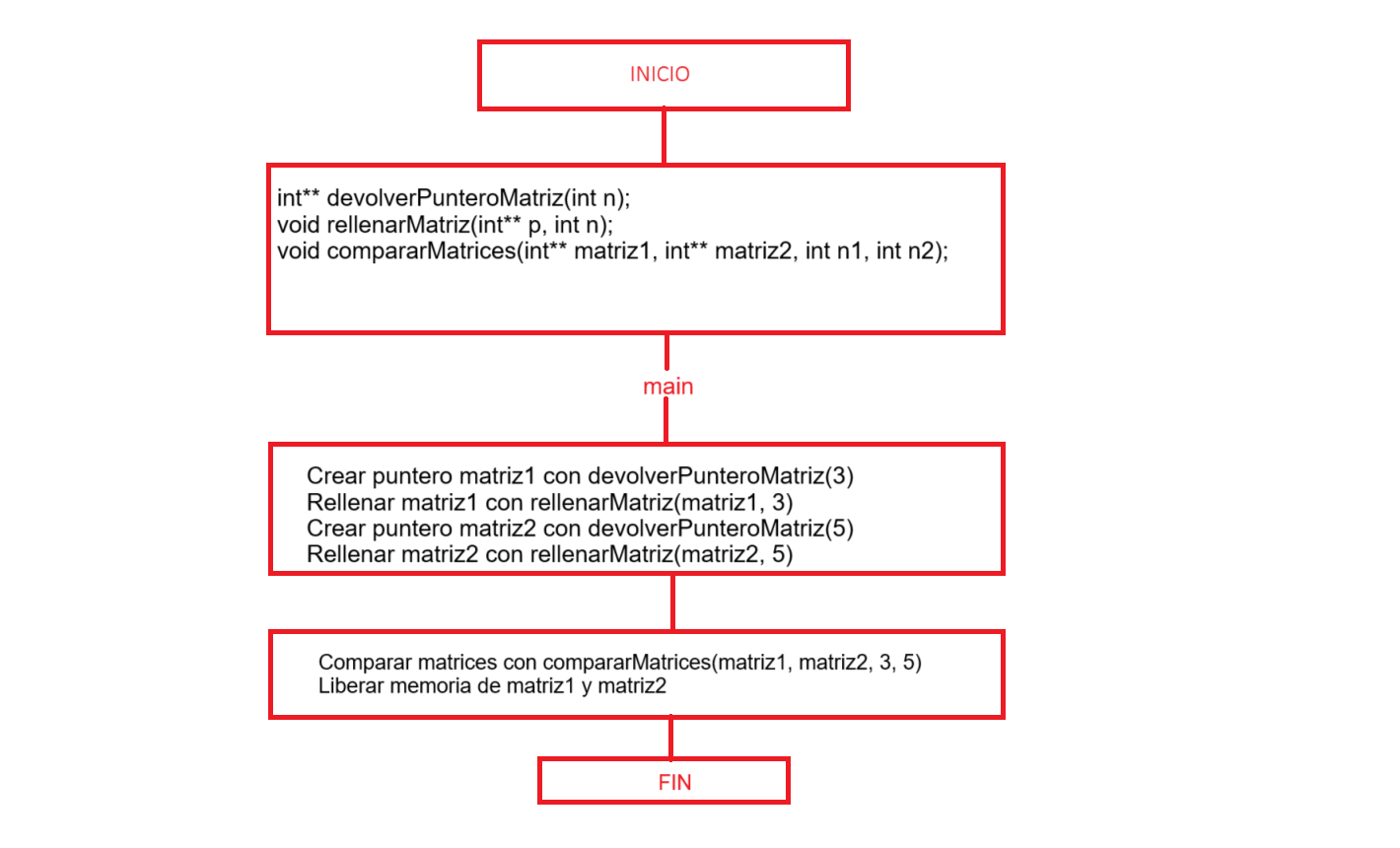


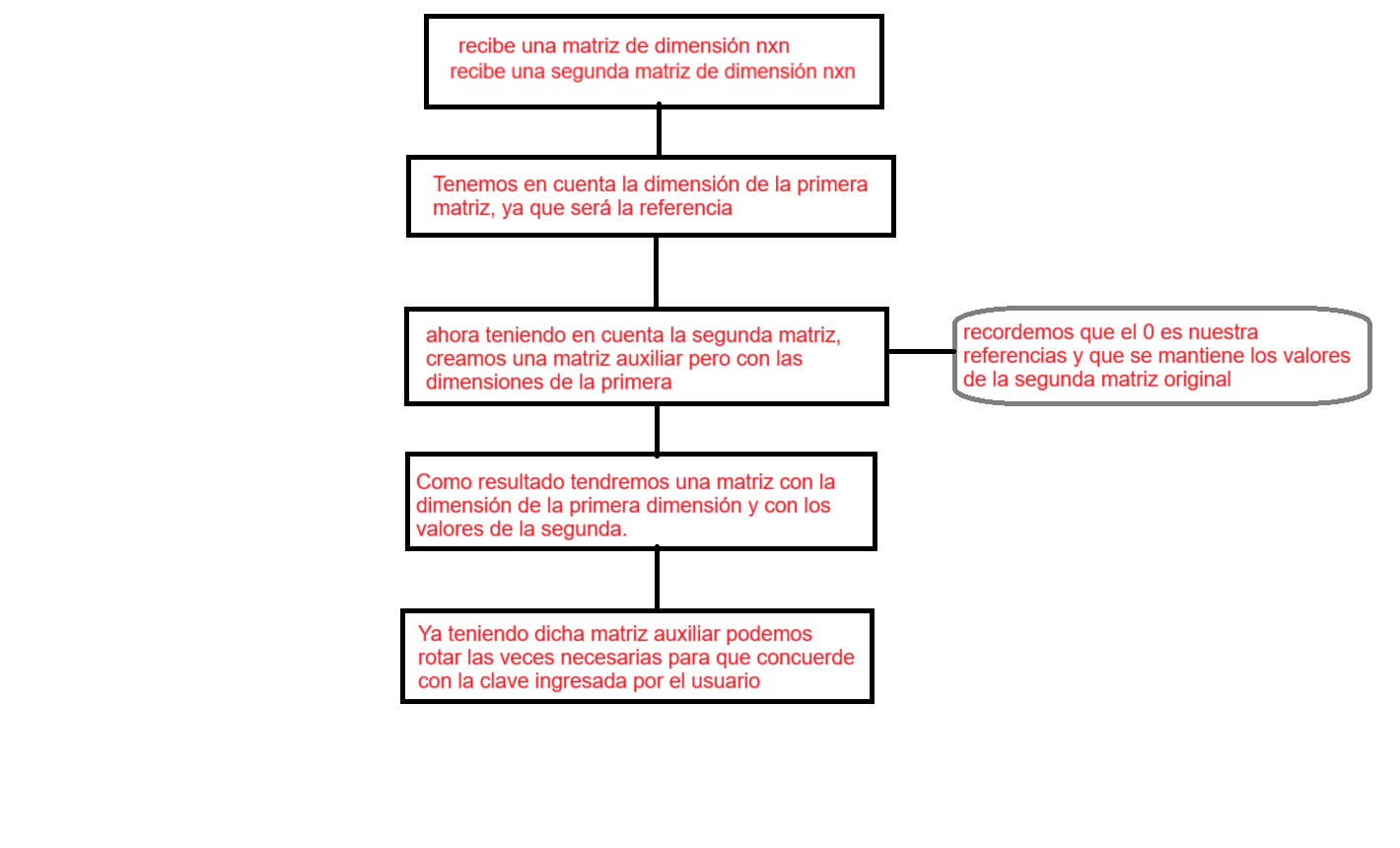
Este es el método de “alinear” las matrices, para después hacer la validación de K.

Es importante recalcar que en este procedimiento se crean dos matrices auxiliares ambas siempre tendrán las dimensiones de la primera matriz y su contenido irá cambiando con base en el par de matrices evaluado.

No se debe dejar pasar por alto que todas las estructuras que pensamos utilizar son completamente dinámicas.

Hasta la fecha 28/03/2024 contemplamos los pasos para desarrollar una alineación entre estructuras M como se describe en el siguiente gráfico.



**Algoritmo de proceso de alineación**

**Estructura de datos**

Para nuestra estructura de datos empezamos definiendo entradas y salidas del programa.

Definimos una entrada para nuestro programa que es:

1. K: es un puntero a un arreglo dinámico cuyo tamaño es N+1 donde N es el número de estructuras M que tendrá la cerradura X, N se infiere a partir de la cantidad de condiciones de comparación que ingresa el usuario reservar un arreglo tipo entero de memoria dinámica culla dirección de memoria del primer elemento será asignada al puntero tipo entero K, para luego pedir línea a línea cada elemento de K, los elementos de K son: k0 la fila de la primera estructura; k1 la columna de la primera estructura; a partir de la segunda posición de K los siguientes elementos corresponden a comparaciones aritméticas entre pares de estructuras M donde se empieza comparando la primera estructura M con la siguiente alineada en X, luego la segunda estructura M con la tercera, hasta llegar a la comparación entre la estructura número N-1 y la estructura número N, los únicos valores posibles para los elementos ki son 1, 0, -1 donde i = 2, 3, …, N+1.

Lo próximo que definimos fueron las salidas de nuestro programa que son:

1. x: Es un apuntador a un arreglo dinámico de tipo unsigned short de tamaño N, donde N es el número de estructuras requeridas por el usuario para la estructura X, este arreglo contiene las dimensiones para cada arreglo dentro de x, por lo tanto, es un requisito para poder crear X, sus elementos son x0, x1, …, xN donde cada elemento debe ser un numero impar positivo, como nuestras estructuras M son de tamaño mxn entonces m = n = xi donde i = 0, 1, …, N.
2. X: Representa una cerradura como tal, por lo tanto, X es un triple puntero a unsigned short que apunta a un arreglo dinámico de dobles punteros a unsigned short, los cuales son las Estructuras M, cada doble puntero a unsigned short apunta a un arreglo dinámico de punteros a unsigned short, que representan las filas de M y por último cada puntero a unsigned short apunta a un arreglo dinámico de unsigned short, que representa las columnas de M. El tamaño del primer arreglo dinámico está dado por N que es el número de estructuras requeridas por el usuario para la cerradura X, el tamaño del segundo arreglo es igual a m que corresponde al número de filas, y el tamaño del último arreglo es n que corresponde al número de columnas.
3. R: Es un apuntador a un arreglo dinámico de tipo unsigned short de tamaño N-1, este arreglo contiene las rotaciones necesarias para cada arreglo M dentro de X, cabe aclarar que es de tamaño N-1, ya que en nuestra propuesta de solución decidimos que la primera M de X nunca rotará, por lo que si en el arreglo al que apunta x el segundo elemento es 5 y en R el primer elemento es 2 esto quiere decir que el segundo arreglo de X que es de tamaño 5x5 deberá estar en su estado número 2 de rotación. Los elementos de R son r0, r1, …, rN-1, además se debe cumplir que 0 <= ri <= 3 donde i = 0, 1, …, N-1.

**Diagrama

Descripción generada automáticamente**

**Algoritmo de generación cerradura**

**Documentación análisis de la clave**

El sistema de apertura de X funciona a partir de la validación de una regla K, que considera el valor de una celda específica, su posición y la ubicación dentro de las diferentes estructuras alineadas. Por ejemplo, una posible entrada K a X(5, 7, 5, 9) puede ser K(4,3,1,-1,1), lo que indica que para la celda ubicada en la fila 4, columna 3 de la primera estructura, su valor A debe ser mayor al de la celda (valor B) de la siguiente estructura con la que se alinea. A su vez el valor B debe ser menor al de la celda (valor C) de la siguiente estructura con la que se alinea. Por último, el valor C debe ser mayor al de la celda (valor D) de la siguiente estructura con la que se alinea.

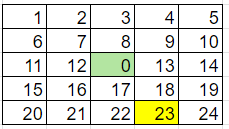
Para poder iniciar el análisis de este punto, lo primero a recalcar es que la columnas y las filsa se van a comenzar a contar desde 0, es decir la matriz 3x3 tiene 0,1,2 filas de arriba hacia abajo respectivamente y 0,1,2 columnas de izquierda a derecha, sabiendo esto podemos ubicarnos correctamente en la matriz inicial.

Ya enfocándonos en las partes de la regla K podemos separarlas por partes

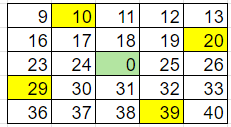
1. Las dos primera entradas de la regla K son las posiciones que queremos comparar, en nuestro ejemplo 4,3 quiere decir que en la 4 fila y 3 columna (en este punto es importante entender que no solo nos da la ubicación que queremos comparar, también nos va decir la dimensión de la matriz mínima inicial) para este ejemplo no podemos iniciar con una matriz 3x3 ya que esta no tiene ni fila 4 ni columna 3, por lo que podemos intuir que debemos iniciar en una matriz 5x5.
2. Las demás entradas de las regla K serán las comparaciones que debemos asegurarnos que se cumplan, entonces si son x comparaciones, obligatoriamente debemos usar x+1 matrices para cubrir esas x comparaciones.
3. Para poder evitar inconvenientes de una solución genérica (con este queremos decir que siempre se use la misma matriz unas y otra vez Ej X(5,5,5,5)) decidimos que no una vez usada una matriz la siguiente matriz no puede ser de la misma dimensión.
4. Usando todas estas especificaciones, ya podemos recorrer el proceso de verificar las regla

Para hacer más fácil la explicación usaremos el mismo ejemplo que nos dan en el documento.

Recibimos del usuario la regla K(4,3,1,-1,1), entonces ya sabemos que la matriz inicial debe ser 5x5

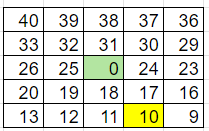


y que la siguiente matriz debe ser de dimensión 7x7 en adelante, primero usaremos la matriz 7x7, alineamos y sacamos la matriz auxiliar

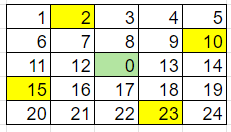


Las celdas coloreadas de amarillo son las posibles comparaciones, en las diferentes rotaciones, y nos damos cuenta que para la primera comparacion (1) nos pide que el valor de la matriz 5x5 en la ubicación 4,3 sea mayor que las de la matriz 7x7 en la misma ubicación, entonces vemos que hay dos posibles rotaciones que nos complen esta regla, cuando es 10 y cuando es 20.Esto no quiere decir que esta mal la opcion 7x7 esto quiere decir que por el momento tenemos dos posibles caminos que recorrer.

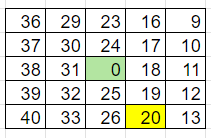
Para la siguiente entrada de la regla (-1) no podemos volver a usar una matriz 7x7 pero si volvemos a tener la opción de la matriz 5x5, entonces la volveremos a usar. Tomaremos el primer camino con la matriz 7x7 y en la ubicación 4,3 va estar el número 10.



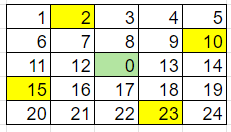
y vemos que la matriz 5x5 tiene otros dos caminos donde cumple la regla (-1) que serían el 15 y el 23



Ahora miremos el segundo camino con la matriz 7x7 y en la ubicación 4,3 va estar el número 20

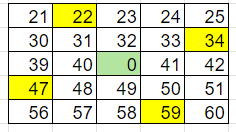


y vemos que la matriz 5x5 solo tiene una opción que cumpla la regla (-1) que sería el 23

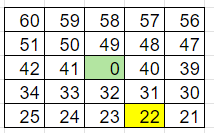
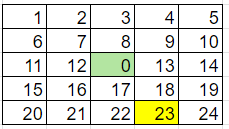
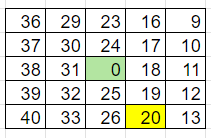
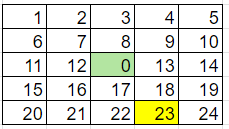


Por lo que lo más apropiado es seguir este segundo camino, entonces como ya dijimos anteriormente no podemos volver a usar una matriz 5x5 pero si podemos volver a usar una matriz 7x7, (a modo de que concuerde con el ejemplo vamos a saltarnos a una matriz 9x9, pero es importante aclarar que volver a coger una matriz 7x7 no estaría mal).

Cogemos la matriz auxiliar de 9x9



y observamos que de los valores posibles en las rotaciones el único que cumple con la regla (1) es la celda con el valor 22. Así que la combinación de matrices que resuelve esta regla K(4,3,1,-1,1).



entonces la cerradura que cumple esta regla es X(5,7,5,9) que es la misma del ejemplo que nos muestra en el documento.