# Preparación de tarea

## Parte I

### Enunciado

En la tarea anterior, vimos cómo se recorre una matriz por fila y por columna. En esta segunda parte, veremos cómo se recorre una matriz por las diagonales. Pero primero, precisa preguntarse, ¿cuántas diagonales tiene la matriz? Para responder esta pregunta, separemos la matriz en distintas secciones.

| **43** | **44** | **45** | **46** | **47** | **48** | **49** |
| --- | --- | --- | --- | --- | --- | --- |
| **42** | **21** | **22** | **23** | **24** | **25** | **26** |
| **41** | **20** | **7** | **8** | **9** | **10** | **27** |
| **40** | **19** | **6** | **1** | **2** | **11** | **28** |
| **39** | **18** | **5** | **4** | **3** | **12** | **29** |
| **38** | **17** | **16** | **15** | **14** | **13** | **30** |
| **37** | **36** | **35** | **34** | **33** | **32** | **31** |

Como bien podemos visualizar, la matriz se compone de dos secciones, a las cuales llamaremos “escalones”. Tanto el escalón inferior como el superior comprenden un total de 6 diagonales cada uno. Si a ambos escalones le sumamos la diagonal principal, que actúa como línea divisoria entre ambos escalones, tenemos un total de 13 diagonales. En su forma general, una matriz espiral contiene un total de diagonales, donde es el tamaño de la matriz cuadrada.

A continuación, procederemos a mostrar cada diagonal que compone a cada escalón.

**Escalón inferior**:

| **Fila** | **Output** |
| --- | --- |
| 1  2  3  4  5  6 | 42 20 6 4 14 32  41 19 5 15 33  40 18 16 34  39 17 35  38 36  37 |

**Escalón superior**:

| **Col** | **Output** |
| --- | --- |
| 1  2  3  4  5  6 | 44 22 8 2 12 30  45 23 9 11 29  46 24 10 28  47 25 27  48 26  49 |

Como quizás se podrá dar cuenta, la manera de visualizar el Output es intencionada, porque asume a los valores de la izquierda como el Input que produce el Output de la derecha.

Por ejemplo, si nos piden producir el segundo *Output* correspondiente al **escalón inferior**, debemos partir desde la Fila 2, y la Columna 0, y en cada paso **aumentamos en uno** las posiciones de **Fila** y **Columna** hasta llegar a la **última Fila** de la matriz. Ahora, si nos piden producir el tercer *Output* correspondiente al **escalón superior**, debemos partir desde la Fila 0 y la Columna 3, y en cada paso, **aumentamos en uno** las posiciones de **Fila** y **Columna** hasta llegar a la **última Columna** de la matriz.

Notar que en ambos escalones es posible recibir 0 como Input y reproducir la **diagonal principal**.

### Problema

Diríjase al servidor de Discord de la carrera, y descargue el código que se subió al canal #tutoria-intro-material. Al igual que en la primera tarea, usted deberá proveer una implementación para los métodos que han sido declarados, pero que carecen de implementación. A continuación, se procederá a documentar lo que hace cada método, para que tenga una idea clara de lo que debe hacer.

*/\*\**

*\* Muestra el contenido de la diagonal perteneciente al* ***escalón inferior*** *especificada por la posición inicial de* ***fila*** *que recibe como parámetro.*

*\**

*\* Si recibe 0 como parámetro, muestra la* ***diagonal principal****.*

*\**

*\* @param* ***matrix*** *La matriz*

*\* @param* ***row*** *La posición inicial de fila*

*\*/*

*void* showLowerDiag(*int* matrix[][**N**], *int* row);

*/\*\**

*\* Muestra el contenido de la diagonal perteneciente al* ***escalón superior*** *especificada por la posición inicial de* ***columna*** *que recibe como parámetro.*

*\**

*\* Si recibe 0 como parámetro, muestra la* ***diagonal principal****.*

*\**

*\* @param* ***matrix*** *La matriz*

*\* @param* ***col*** *La posición inicial de columna*

*\*/*

*void* showUpperDiag(*int* matrix[][**N**], *int* col);

*/\*\**

*\* Muestra diagonal a diagonal el contenido de* ***todas las diagonales*** *que componen al* ***escalón inferior****. Si el parámetro booleano es verdadero, la primera diagonal en ser mostrada es la* ***diagonal principal****.*

*\**

*\* @param* ***matrix*** *La matriz*

*\* @param* ***showMainDiag*** *Un valor booleano que indica si la primera diagonal en ser mostrada debería ser la diagonal principal.*

*\*/*

*void* traverseLowerEchelon(*int* matrix[][**N**], **bool** showMainDiag)

*/\*\**

*\* Muestra diagonal a diagonal el contenido de* ***todas las diagonales*** *que componen al* ***escalón superior****. Si el parámetro booleano es verdadero, la primera diagonal en ser mostrada es la* ***diagonal principal****.*

*\**

*\* @param* ***matrix*** *La matriz*

*\* @param* ***showMainDiag*** *Un valor booleano que indica si la primera diagonal en ser mostrada debería ser la diagonal principal.*

*\*/*

*void* traverseUpperEchelon(*int* matrix[][**N**], **bool** showMainDiag);

## 

## Parte II

### Enunciado

Los escalones mostrados a continuación originan a partir de la diagonal invertida principal. Al igual que en el caso anterior, tenemos un total de 13 diagonales, con la diagonal invertida principal actuando como línea divisoria entre ambos escalones.

| **43** | **44** | **45** | **46** | **47** | **48** | **49** |
| --- | --- | --- | --- | --- | --- | --- |
| **42** | **21** | **22** | **23** | **24** | **25** | **26** |
| **41** | **20** | **7** | **8** | **9** | **10** | **27** |
| **40** | **19** | **6** | **1** | **2** | **11** | **28** |
| **39** | **18** | **5** | **4** | **3** | **12** | **29** |
| **38** | **17** | **16** | **15** | **14** | **13** | **30** |
| **37** | **36** | **35** | **34** | **33** | **32** | **31** |

A continuación, procederemos a mostrar cada diagonal que compone a cada escalón.

**Escalón inferior**:

| **Fila** | **Output** |
| --- | --- |
| 0  1  2  3  4  5 | 26 10 2 4 16 36  27 11 3 15 35  28 12 14 34  29 13 33  30 32  31 |

El patrón de recorrido de este escalón es el siguiente: empieza desde **la fila indicada** como Input y desde **la última columna**, y en cada paso **aumenta en uno la fila** y **disminuye en uno la columna** hasta llegar a la **última fila**.

**Escalón superior**:

| **Col** | **Output** |
| --- | --- |
| 5  4  3  2  1  0 | 48 24 8 6 18 38  47 23 7 19 39  46 22 20 40  45 21 41  44 42  43 |

El patrón de recorrido de este escalón es el siguiente: empieza desde la **primera fila** y desde **la columna indicada** como Input, y en cada paso **aumenta en uno la fila** y **disminuye en uno la columna** hasta llegar a la **primera columna**.

Como bien se podrá apreciar, la forma de recorrer cada escalón es exactamente el mismo, excepto por la condición de término, que es hasta llegar a la **última fila**, en el caso del escalón inferior, y a la **primera columna**, en el caso del escalón superior.

Notar que en ambos escalones es posible reproducir la **diagonal inversa principal**. Para que esto suceda, deberá recibir los Inputs 0 y N - 1 correspondientes al escalón inferior y superior, respectivamente.

### Problema

Al igual que en la sección anterior, usted deberá proveer una implementación para los métodos que han sido declarados, pero que carecen de implementación.

*/\*\**

*\* Muestra el contenido de la* ***diagonal invertida*** *perteneciente al* ***escalón inferior*** *especificada por la posición inicial de* ***fila*** *que recibe como parámetro.*

*\**

*\* Si recibe 0 como parámetro, muestra la* ***diagonal principal****.*

*\**

*\* @param* ***matrix*** *La matriz*

*\* @param* ***row*** *La posición inicial de fila.*

*\*/*

*void* showLowerInvertedDiag(*int* matrix[][**N**], *int* row);

*/\*\**

*\* Muestra el contenido de la* ***diagonal invertida*** *perteneciente al* ***escalón superior*** *especificada por la posición inicial de* ***columna*** *que recibe como parámetro.*

*\**

*\* Si recibe N - 1 como parámetro, muestra la* ***diagonal principal****.*

*\**

*\* @param* ***matrix*** *La matriz*

*\* @param* ***col*** *La posición inicial de columna.*

*\*/*

*void* showUpperInvertedDiag(*int* matrix[][**N**], *int* col);

*/\*\**

*\* Muestra diagonal a diagonal el contenido de* ***todas las diagonales invertidas*** *que componen al* ***escalón inferior****. Si el parámetro booleano es verdadero, la primera diagonal invertida en ser mostrada es la* ***diagonal invertida principal****.*

*\**

*\* @param* ***matrix*** *La matriz*

*\* @param* ***showMainDiag*** *Un valor booleano que indica si la primera diagonal invertida en ser mostrada debería ser la diagonal invertida principal.*

*\*/*

*void* traverseLowerInvertedEchelon(*int* matrix[][**N**], **bool** showMainDiag);

*/\*\**

*\* Muestra diagonal a diagonal el contenido de* ***todas las diagonales invertidas*** *que componen al* ***escalón superior****. Si el parámetro booleano es verdadero, la primera diagonal invertida en ser mostrada es la* ***diagonal invertida principal****.*

*\**

*\* @param* ***matrix*** *La matriz*

*\* @param* ***showMainDiag*** *Un valor booleano que indica si la primera diagonal invertida en ser mostrada debería ser la diagonal invertida principal.*

*\*/*

*void* traverseLowerInvertedEchelon(*int* matrix[][**N**], **bool** showMainDiag);