

# Laboratorio 2 - Operaciones matriciales

INS125 - Leguajes de programación  
Universidad Andrés Bello

23 de abril de 2020

## 1. Matrices

Las Matrices son un arreglo bidimensional de números, utilizadas comúnmente en computación para almacenar información y realizar cálculos matemáticos. También suelen representar imágenes.

A continuación describiremos dos operaciones fundamentales que se aplican sobre matrices.

## 2. Suma de matrices

La suma de dos matrices, denotada como  $A+B$ , consiste en sumar, elemento a elemento, cada una de las componentes de una matriz  $A$  con una matriz  $B$ . Para ello ambas matrices deben tener la misma dimensión, es decir, deben tener igual número de filas  $M$  y columnas  $N$ . A continuación es posible observar la definición de esta operación.

$$A+B = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1n} \\ b_{21} & b_{22} & \dots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{m1} & b_{m2} & \dots & b_{mn} \end{bmatrix} = \begin{bmatrix} a_{11} + b_{11} & a_{12} + b_{12} & \dots & a_{1n} + b_{1n} \\ a_{21} + b_{21} & a_{22} + b_{22} & \dots & a_{2n} + b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} + b_{m1} & a_{m2} + b_{m2} & \dots & a_{mn} + b_{mn} \end{bmatrix}$$

A continuación, es posible observar un ejemplo de la suma de dos matrices de dimensión  $3 \times 2$  y  $3 \times 3$ :

$$\begin{bmatrix} 1 & 3 \\ 1 & 0 \\ 1 & 2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 7 & 5 \\ 2 & 2 \end{bmatrix} = \begin{bmatrix} 1+0 & 3+0 \\ 1+7 & 0+5 \\ 1+2 & 2+1 \end{bmatrix} = \begin{bmatrix} 1 & 3 \\ 8 & 5 \\ 3 & 3 \end{bmatrix} \quad (1)$$

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} + \begin{bmatrix} 10 & 11 & 12 \\ 13 & 14 & 15 \\ 16 & 17 & 18 \end{bmatrix} = \begin{bmatrix} 1+10 & 2+11 & 3+12 \\ 4+13 & 5+14 & 6+15 \\ 7+16 & 8+17 & 9+18 \end{bmatrix} = \begin{bmatrix} 11 & 13 & 15 \\ 17 & 19 & 21 \\ 23 & 25 & 27 \end{bmatrix} \quad (2)$$

### 3. Convolución de Matrices

Es un dominio de las matemáticas utilizado usualmente en los filtros de imágenes. Este filtro se encarga de realizar una operación matemática a cada elemento de la matriz. Los filtros más utilizados son matrices de tamaño  $3 \times 3$  o  $5 \times 5$ .

Consiste en examinar cada posición de la matriz. Cada posición con sus alrededores son multiplicados por el filtro entregado y sumados. Este resultado se guarda en una nueva matriz en la posición examinada.

Para esta tarea la convolución de matrices se realizará para matrices cuadradas con un largo  $N \geq 3$  a la cual se le aplicara un filtro de tamaño  $3 \times 3$ . Además se omite el cálculo para el borde de la matriz. El **resultado de la Convolución** es una matriz cuadrada de un largo  $N - 2$

#### 3.1. Ejercicio Convolución

Si consideramos una Matriz de  $4 \times 4$

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix} \quad (3)$$

Y un filtro Gaussiano de  $3 \times 3$

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (4)$$

Como el borde de la matriz no se debe considerar en este caso, solo debemos considerar los siguiente valores para aplicar la convolución, pero en su cálculo debe ser con la matriz completa.

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix} \Rightarrow \begin{bmatrix} 6 & 7 \\ 10 & 11 \end{bmatrix} \quad (5)$$

Como nuestro filtro es de  $3 \times 3$  debemos ir creando distintas submatrices de  $3 \times 3$  de la matriz principal. En este caso

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 2 & 3 \\ 5 & 6 & 7 \\ 9 & 10 & 11 \end{bmatrix} \times \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (6)$$

Convolución del primer valor

$$1 * 0 + 2 * 1 + 3 * 0 + 5 * 1 + 6 * 0 + 7 * 1 + 9 * 0 + 10 * 1 + 11 * 0 = 24$$

Donde el primer resultado es 24

$$\begin{bmatrix} 24 & ? \\ ? & ? \end{bmatrix}$$

Segundo valor

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix} \Rightarrow \begin{bmatrix} 2 & 3 & 4 \\ 6 & 7 & 8 \\ 10 & 11 & 12 \end{bmatrix} \times \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (7)$$

Convolución del segundo valor

$$2 * 0 + 3 * 1 + 4 * 0 + 6 * 1 + 7 * 0 + 8 * 1 + 10 * 0 + 11 * 1 + 12 * 0 = 28$$

Donde el primer resultado es 28

$$\begin{bmatrix} 24 & 28 \\ ? & ? \end{bmatrix}$$

Tercer valor

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix} \Rightarrow \begin{bmatrix} 5 & 6 & 7 \\ 9 & 10 & 11 \\ 13 & 14 & 15 \end{bmatrix} \times \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (8)$$

Convolución del tercer valor

$$5 * 0 + 6 * 1 + 7 * 0 + 9 * 1 + 10 * 0 + 11 * 1 + 13 * 0 + 14 * 1 + 15 * 0 = 40$$

Donde el resultado es 40

$$\begin{bmatrix} 24 & 28 \\ 40 & ? \end{bmatrix}$$

Cuarto valor

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix} \Rightarrow \begin{bmatrix} 6 & 7 & 8 \\ 10 & 11 & 12 \\ 14 & 15 & 16 \end{bmatrix} \times \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (9)$$

Convolución del cuarto valor

$$6 * 0 + 7 * 1 + 8 * 0 + 10 * 1 + 11 * 0 + 12 * 1 + 14 * 0 + 15 * 1 + 16 * 0 = 44$$

Donde el resultado es 44 y completando la convolución

$$\begin{bmatrix} 24 & 28 \\ 40 & 44 \end{bmatrix}$$

## 4. Tarea

En esta tarea, a usted se le solicita implementar una suma de matrices y una Convolución de matrices utilizando punteros dobles. Las matrices contendrán números existentes en un archivo de texto separados por “;” (punto y coma) .

Usted recibirá dos matrices cuadradas del mismo tamaño  $A$  y  $B$ ; la cantidad de filas y columnas  $N$ ; y una matriz filtro  $F$  de tamaño  $3 \times 3$  para emplear en la Convolución.

Para estas operaciones, usted debe utilizar las funciones estructuradas en el modulo `matriz.h` la cual contiene las funciones correspondientes y los argumentos.

Por lo tanto, para sumar dos matrices usted debe utilizar la función:

```
1 int ** Suma(int **matrizA , int **matrizB , int tamano)
```

Luego al resultado de la suma debe aplicar la Convolución dado el filtro entregado. Para operar la convolución usted deberá llamar a la función:

```
■ int ** Convolucion(int **matrizSumada , int **filtro , int tamano)
```

Como salida, su programa debe escribir dos archivos. El primer archivo de salida debe contener el resultado de la suma de matrices, mientras que el segundo archivo el resultado de la Convolución. Ambos archivo de salida deben tener una cantidad de  $N$  líneas, donde cada línea debe guardar la fila completa de cada matriz separando los números por “;”.

Por lo tanto, como entrada su programa recibirá tres archivos donde:

- Dos corresponden a la matriz los cuales contendrán  $N$  líneas
- Uno corresponde al filtro  $F$  que tendrá 3 líneas.

El nombre y/o ruta de los archivos de entrada y salida, como el tamaño de las matrices deben ser entregados por parámetros de entrada.

### 4.1. Parámetros de entrada

Las instrucciones se darán por parámetros al momento de ejecutar su programa, para ello se dispondrán una serie de comandos donde cada uno de ellos viene seguido con su valor correspondiente. Se recomienda ver [Getopt](#). **Esto ya se encuentra implementado en el código base provisto.**

- **-a:** Indica el nombre del archivo de entrada de la primera matriz.
- **-b:** Indica el nombre del archivo de entrada de la segunda matriz.
- **-l:** Indica numero de filas de la la matriz. (Como es una matriz cuadrada el numero de filas es igual al de columnas).
- **-f:** Indica el nombre del archivo de entrada para el filtro a utilizar (una matriz de  $3 \times 3$ ).
- **-s:** Nombre del archivo de salida de la Suma de matrices.
- **-c:** Nombre del archivo de salida de la Convolución.

## 4.2. Ejemplo de parámetros de entrada

### 4.2.1. Ejemplo parámetros 1

```
$ ./main -a mUno.in -b mDos.in -l 3 -f filG.in -s suma.out -c conv.out
```

- **mUno.in:** Nombre y/o ruta del archivo de entrada para la matriz  $A$ .
- **mDos.in:** Nombre y/o ruta del archivo de entrada para la matriz  $B$ .
- **3:** Numero de filas y/o columnas de las matrices  $A$  y  $B$ .
- **filtroG.in:** Nombre y/o ruta del archivo de entrada filtro  $F$  (una matriz de  $3 \times 3$ ).
- **suma.out:** Nombre y/o ruta del archivo de salida para la Suma de matrices.
- **conv.out:** Nombre y/o ruta del archivo de salida para la Convolución.

### 4.2.2. Ejemplo parametros 2

```
$ ./main -a mA.in -b mB.in -l 5 -f filC.in -s mSuma.out -c mConv.out
```

- **mA.in:** Nombre y/o ruta del archivo de entrada para la matriz  $A$
- **mB.in:** Nombre y/o ruta del archivo de entrada para la matriz  $B$
- **5:** Numero de filas y/o columnas de las matrices  $A$  y  $B$
- **filtroC.in:** Nombre y/o ruta del archivo de entrada filtro  $F$  (una matriz de  $3 \times 3$ )
- **mSuma.out:** Nombre y/o ruta del archivo de salida para la Suma de matrices.
- **mConv.out:** Nombre y/o ruta del archivo de salida para la Convolución.

## 5. Ejemplo de archivos de entrada y salida

### 5.1. Ejemplo 1

En este ejemplo, se entregan las matrices  $A$ ,  $B$  y  $F$  con un largo de  $N = 3$

#### 5.1.1. Archivos de entrada

Matriz  $A$

1;2;3

4;5;6

7;8;9

Matriz  $B$

10;11;12

13;14;15

16;17;18

Filtro  $F$

0;1;0

1;0;1

0;1;0

#### 5.1.2. Archivos de salida

Matriz Suma

11;13;15

17;19;21

23;25;27

Matriz Convolución

76

## 5.2. Ejemplo 2

En este ejemplo, se entregan las matrices  $A$ ,  $B$  y  $F$  con un largo de  $N = 4$

### 5.2.1. Archivos de entrada

Matriz  $A$

```
1;2;3;4
5;6;7;8
9;10;11;12
13;14;15;16
```

Matriz  $B$

```
0;0;0;0
0;0;0;0
0;0;0;0
0;0;0;0
```

Filtro  $F$

```
0;1;0
1;0;1
0;1;0
```

### 5.2.2. Archivos de salida

Matriz Suma

```
1;2;3;4
5;6;7;8
9;10;11;12
13;14;15;16
```

Matriz Convolución

```
24;28
40;44
```

## 6. Instrucciones

- Fecha de entrega: Jueves 7 de mayo, 2020 a las 23:59.
- Método de entrega: Su repositorio privado creado a través del link de la tarea. **No se aceptaran entregas en repositorios no creados a través del link de la tarea.**
- Trabajo personal hecho en lenguaje C.
- Para comenzar su tarea, clone su repositorio y utilice el archivo `main.c`, `matriz.c` y `matriz.h`, con código pre hecho. Ese código le servirá recibir los parámetros y para separar los números en el archivo de entrada. Puede editarlo con libertad.
- Su repositorio de la tarea debe contener tres archivos los cuales contendrá su programa, y estos son llamados `main.c`, `matriz.c` y `matriz.h`. No incluya los archivos de entrada o salida en su repositorio. Utilice un archivo `.gitignore` para no subirlo a su repositorio remoto.
- Su programa debe recibir y generar los archivos entrada y salida que indique la entrada por parámetros.
- Solo puede utilizar bibliotecas estándar. Verifique que su programa compila al ejecutar el comando `gcc main.c matriz.c -o main.o`.
- El proceso de revisión será automatizado. Es importante respetar el formato establecido para los archivos de entrada y salida. Este formato no es modificable. Por lo tanto, si su archivo de salida no corresponde al formato preestablecido, su calificación será deficiente.
- Las preguntas sobre la tarea deben ser formuladas como un Issue en el repositorio del laboratorio ubicado en el siguiente link <https://github.com/INS125/Laboratorio/issues>

## 7. Recomendaciones

- Se recomienda hacer commits parciales. Si su archivo de salida no contiene toda la matriz o el resultado completo de la Convolución pero si contiene parcialmente la matriz, puede obtener puntaje parcial.
- Si su programa no crea el o los archivos de salida, o no compila, será evaluado con la nota mínima.
- En el repositorio oficial del laboratorio puede encontrar dos ejemplos de archivos de entrada y su correspondiente salida. <https://github.com/INS125/Laboratorio/>
- Recuerde solicitar unirse a github student. Si no lo hace, no podrá hacer su código privado y cualquiera podría visualizar su tarea.
- Es un trabajo personal. Es su responsabilidad cuidar su tarea.



## 8. Código de honor

Toda persona inscrita en este curso se compromete a:

- Actuar con honestidad, rectitud y buena fe frente a sus profesores y compañeros.
- No presentar trabajos o citas de otras personas como propias o sin su correspondiente citación, ya sea de algún compañero, libro o extraídos de internet como también a no reutilizar trabajos presentados en semestres anteriores como trabajos originales.
- No copiar a compañeros ni hacer uso de ayudas o comunicaciones fuera de lo permitido durante las evaluaciones.

Cualquier alumno o alumna que no respete el código de honor durante una evaluación (sea este la entrega de una tarea o el desarrollo de una prueba o control tanto durante la cátedra como el laboratorio) será evaluado con la nota mínima y será virtud de profesor, de acuerdo con la gravedad de la falta, las acciones siguientes a tomar.

## 9. Anexo

### 9.1. Convolución 3x3

Si consideramos una Matriz de  $3 \times 3$

$$\begin{bmatrix} 11 & 13 & 15 \\ 17 & 19 & 21 \\ 23 & 25 & 27 \end{bmatrix} \quad (10)$$

Y un filtro Gaussiano de  $3 \times 3$

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (11)$$

Como el borde de la matriz no se debe considerar en este caso, solo debemos considerar los siguiente valores para aplicar la convolución, pero en su calculo debe ser con la matriz completa.

$$\begin{bmatrix} 11 & 13 & 15 \\ 17 & 19 & 21 \\ 23 & 25 & 27 \end{bmatrix} \Rightarrow [19] \quad (12)$$

Como nuestro filtro es de  $3 \times 3$  debemos ir creando distintas submatrices de  $3 \times 3$  de la matriz principal. En este caso

$$\begin{bmatrix} 11 & 13 & 15 \\ 17 & 19 & 21 \\ 23 & 25 & 27 \end{bmatrix} \Rightarrow \begin{bmatrix} 11 & 13 & 15 \\ 17 & 19 & 21 \\ 23 & 25 & 27 \end{bmatrix} \times \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (13)$$

Convolución de la única sección

$$11 * 0 + 13 * 1 + 15 * 0 + 17 * 1 + 19 * 0 + 21 * 1 + 23 * 0 + 25 * 1 + 27 * 0 = 76$$

Donde el resultado es 76

$$[76]$$