

<b>Nombre de la práctica</b>	<b>ORGANIZACIÓN DE DATOS (UNIDAD 4)</b>			<b>No.</b>	<b>3</b>
<b>Asignatura:</b>	<b>FUNDAMENTOS DE PROGRAMACIÓN</b>	<b>Carrera:</b>	<b>INGENIERÍA EN SISTEMAS COMPUTACIONALES</b>	<b>Duración de la práctica (Hrs)</b>	<b>10 horas</b>

**NOMBRE DEL ALUMNO:** Vanessa Fabiola Hernández Ramírez

**GRUPO:** 3101

## I. Competencia(s) específica(s):

Conoce y aplica estructuras de datos en un lenguaje de programación que permitan la organización de datos en la resolución de problemas reales.

**Encuadre con CACEI:** Registra el (los) atributo(s) de egreso y los criterios de desempeño que se evaluarán en esta práctica.

No. atributo	Atributos de egreso del PE que impactan en la asignatura	No. Criterio	Criterios de desempeño	No. Indicador	Indicadores
1	El estudiante identificará los principios de las ciencias básicas para la resolución de problemas prácticos de ingeniería	CD1	Identifica problemas relacionados con aplicación de la ingeniería	I1	Análisis de problemas y/o necesidades
				I2	Empleo herramientas para el análisis
		CD2	Propone alternativas de solución	I1	Diseño algorítmico
				I3	Modelado de programas
				I4	Uso de metodologías
		CD3	Analiza y comprueba los resultados generados	I1	Comprobación de resultados
I2	Toma de decisiones				
3	El estudiante plantea soluciones basadas en tecnologías empleando su juicio ingenieril para valorar necesidades, recursos y resultados esperados.	CD1	Emplea los conocimientos adquiridos para el desarrollar soluciones	I1	Elección de metodologías, técnicas y/o herramientas para el desarrollo de soluciones
				I2	Uso de metodologías adecuadas para el desarrollo de proyectos
				I3	Generación de productos y/o proyectos
		CD2	Analiza y comprueba resultados	I1	Realizar pruebas a los productos obtenidos
				I2	Documentar información de las pruebas realizadas y los resultados

## II. Lugar de realización de la práctica (laboratorio, taller, aula u otro):

Laboratorio de cómputo y equipo de cómputo personal.

## III. Material empleado:

- Equipo de cómputo
- Netbeans - jdk

## IV. Desarrollo de la práctica:

### UNIDAD 4

#### PROGRAMA 1 :

**A) Descripción del problema:** El tecnológico de Jilotepec ha solicitado un programa para hacer más rápido y fácil la capturasión de datos sobre las calificaciones del alumno de primer semestre, en el cual te muestre las calificaciones de cada unidad, de cada materia, el promedio final de cada materia y por último la calificación final del semestre

#### B) Código de java

```
1  import javax.swing.JOptionPane;
2  public class ArrglosEjem {
3      public static void main (String[] args) {
4          //Declaracion de variables1
5          String materias[]={ "CALCDIF", "FUNDPRO", "FUNDINV", "QUIMICA", "DESASUS",
6              "MATEDIS", "TUTORIA", };
7          String encabezado[]={ "MATERIA", "U1", "U2", "U3", "U4", "U5", "PromFinal" };
8          byte calificacion[] [] = new byte[7][6];
9          byte x, y;
10         double prom;
11         double PromFinal;
12         //ingresar calificaciones de las materias
13         for (x = 0; x <= 5; x++) { //filas(materia)
14             for (y = 0; y < 5; y++) { //columnas(unidad)
15                 calificacion[x][y] = Byte.parseByte(JOptionPane.showInputDialog("Ingresa"
16                     + " la calificacion de: " + materias[x] + " Unidad" + (y+1)));
17             }
18         }
19
20     }
21     //Impresion de calificaciones
22     for (x = 0; x <= 6; x++) {
```

```
23 System.out.print(encabezado[x]+"\\t");
24 }
25 System.out.println("");
26 PromFinal=0;
27 for (x = 0; x <=5; x++) { //filas(materia)
28     prom=0;
29     System.out.print(materias[x]+"\\t");
30     for (y = 0; y < 5; y++) { //columnas(unidad)
31         System.out.print(calificacion[x][y]+"\\t");
32         prom = (prom + calificacion[x][y]);
33     }
34     System.out.print(prom/5);
35     PromFinal= (PromFinal + prom);
36     System.out.println("");
37 }
38
39 System.out.print("Tu promedio final de semestre es: "+PromFinal/7);
40
41 }
42
43
44
```

### C) Pantallas resultantes-probando con distintos datos

Entrada

Ingresa la calificacion de: CALCDIF Unidad1

80

Aceptar Cancelar

Entrada

Ingresa la calificacion de: FUNDPRO Unidad1

90

Aceptar Cancelar

Entrada

Ingresa la calificación de: FUNDINV Unidad1

100

Aceptar Cancelar

Entrada

Ingresa la calificación de: QUIMICA Unidad1

90

Aceptar Cancelar

Entrada

Ingresa la calificación de: DESASUS Unidad1

90

Aceptar Cancelar

Entrada

Ingresa la calificación de: DESASUS Unidad2

90

Aceptar Cancelar

Entrada

Ingresa la calificación de: MATEDIS Unidad2

90

Aceptar Cancelar

## Tabla de resultados

Output - U4\_3101\_VanessaFabiola3 (run)

```
run:
MATERIA U1      U2      U3      U4      U5      PromFinal
CALCDIF 90      89      89      78      89      87.0
FUNDPRO 90      90      100     90      89      91.8
FUNDINV 90      89      78      100     89      89.2
QUIMICA 90      75      89      90      89      86.6
DESASUS 100     100     89      78      90      91.4
MATEDIS 90      100     90      89      89      91.6
Tu promedio final de semestre es: 384.0BUILD SUCCESSFUL (total
```

## D) Que aprendizaje específico lograste con este ejercicio

El uso de este código que su principal función fue el uso del algoritmo llamado matriz, el cual fue un elemento fundamental dentro del código. A través de este enfoque, hemos aprendido a estructurar y gestionar grandes volúmenes de información de manera eficiente, lo que no solo facilita el acceso a los datos, sino que también optimiza la toma de decisiones basadas en algún tipo de información que deseamos ingresar o bien datos que necesitamos.

Este ejercicio nos ha permitido comprender la importancia de las herramientas matemáticas y tecnológicas en la educación, y cómo, al aplicar métodos estructurados como las matrices, podemos obtener resultados precisos y organizados con ayuda de la aplicación de matrices que dentro del programa de java este es un importante elemento.

## PROGRAMA 2 :

**A) Descripción del problema:** Elabora un programa donde dos matrices al ser multiplicadas den una tercera con el resultado de tal multiplicación, sin embargo, el primer dígito de la primera matriz se multiplicará con el último de la segunda matriz y así sucesivamente todos los dígitos de las matrices.

## B) Código de java

```
1  import javax.swing.JOptionPane;
2  public class Mult_Matrices {
3      public static void main (String[] args) {
4          String filas[]={"0","1","2"};
5          String encabezado[]={" ","0","1","2"};
6          byte numero[][]=new byte[4][3];
7          byte numero2[][]=new byte[4][3];
8          float v3[][]=new float[4][3];
9          byte x,y,n,s;
10         //ingresar los numeros
11         for (x = 0; x < 3; x++) { //filas
12             for (y = 0; y < 3; y++) { //encabezado
13                 numero[x][y]= Byte.parseByte(JOptionPane.showInputDialog("Ingresa un numero"));
14             }
15         }
16         //Impresion
17         for(x = 0; x <= 3; x++){
18             System.out.print("\033[35m"+encabezado[x]+"\\t");
19         }
20         System.out.println("");
21         for (x = 0; x < 3; x++) { //filas
```

```
23     System.out.print(filas[x]+ "\t");
24     for (y = 0; y < 3; y++) { //encabezado
25         System.out.print(numero[x][y]+ "\t" );
26
27     }
28     System.out.println("");
29 }
30 System.out.println("\n");
31 //ingresar los numeros
32 for (x = 0; x < 3; x++) { //filas
33     for (y = 0; y < 3; y++) { //encabezado
34         numero2[x][y]= Byte.parseByte(JOptionPane.showInputDialog("Ingresa un "
35             + "segundo numero"));
36     }
37
38 }
39 //Impresion
40 for(x = 0; x <= 3; x++){
41     System.out.print("\033[35m"+encabezado[x]+ "\t");
42 }
43 System.out.println("");
44 for (x = 0; x < 3; x++) { //filas
45     System.out.print(filas[x]+ "\t");
46     for (y = 0; y < 3; y++) { //encabezado
47         System.out.print(numero2[x][y]+ "\t" );
48
49     }
50
51     System.out.println("");
52 }
53 System.out.println("\n");
54 //calculacion de datos
55 n=2;
56 for (x = 0; x < 3; x++) { //filas
57     s=2;
58     for (y = 0; y < 3; y++) { //encabezado
59         v3[x][y]= numero[x][y] * numero2[n][s]; //incremento ([x][y]) -decremento [n][s]
60         s--;
61     }
62     n--;
63 }
64 //Impresion
65 for(x = 0; x <= 3; x++){
66     System.out.print("\033[35m"+encabezado[x]+ "\t");
```

```
66 System.out.print("\033[35m"+encabezado[x)+"\t");
67 }
68 System.out.println("");
69 for (x = 0; x < 3; x++) { //filas
70     System.out.print(filas[x]+ "\t");
71     for (y = 0; y < 3; y++) { //encabezado
72         System.out.print(v3[x][y)+"\t" );
73     }
74     System.out.println("");
75 }
76 }
77 }
78 }
```

## c) Pantallas resultantes-probando con distintos datos

Entrada

Ingresa un numero

9

Aceptar Cancelar

Output - U4\_3101\_VanessaFabiola3 (run)

```
run:
0      0      1      2
0      9      12     3
1      5      6      8
2      9      7      2
```

Entrada

Ingresa un segundo numero

Aceptar Cancelar

Output - U4\_3101\_VanessaFabiola3 (run)

```
run:
      0      1      2
0      1      2      3
1      4      5      6
2      7      8      9

      0      1      2
0      1      2      3
1      4      5      6
2      7      8      9

      0      1      2
0      9.0    16.0    21.0
1     24.0    25.0    24.0
2     21.0    16.0     9.0

BUILD SUCCESSFUL (total time: 1 minute 0 seconds)
```

## D) Que aprendizaje específico lograste con este ejercicio

Entender cómo se calculan los elementos de la matriz y el producto a partir de las matrices originales. Específicamente, cómo cada elemento resulta de la suma de productos entre los elementos desde la primera matriz y los elementos de la segunda, Identificar las condiciones para que dos matrices sean multiplicables siendo estas que desde el ultimo dígito de la segunda se multipliquen con el primer dígito de la primera, así obteniendo la tercera matriz con dichos resultados. Para esto podemos hacer uso de ciertos elementos como lo fue el uso de bucles “for” y de incrementos y decrementos.



## V. Conclusiones:

El uso de matrices en programación es fundamental, ya que permiten almacenar y manipular colecciones de datos de manera eficiente y organizada. En Java, las matrices son estructuras de datos muy útiles que pueden representar datos en una tabla, una cuadrícula.

En términos de sintaxis, las matrices en Java son objetos, lo que significa que poseen propiedades y métodos propios, aunque su tamaño debe definirse al momento de crearlas. Se pueden declarar, inicializar y manipular con facilidad, pero es importante gestionar correctamente su uso. Las matrices en Java son estructuras de datos esenciales para almacenar y organizar colecciones de datos de manera eficiente. Permiten representar y manipular información en una sola variable, facilitando el trabajo con grandes volúmenes de datos relacionados, como en tablas, imágenes o simulaciones. A través de su uso, se aprende a manejar índices correctamente y gestionar datos de manera más estructurada. El dominio de las matrices en Java es clave para resolver problemas complejos y optimizar el rendimiento en aplicaciones reales. En resumen, las matrices son fundamentales para mejorar la eficiencia y organización en la programación.

El uso de bucles fue fundamental para la acomodación de datos de la matriz de cada ejercicio, gracias a ello podemos decir que fue la parte más laboriosa, saber en dónde iban exactamente los incrementos y decrementos para así obtener el resultado esperado, pero bien se obtuvo el resultado esperando viendo como este simple uso puede llevarte a obtener grandes resultados acumulados en uno solo.

En resumen, las matrices en Java son herramientas poderosas y versátiles, esenciales para manejar datos de manera eficiente en una variedad de aplicaciones.