



## Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

# Laboratorios de computación salas A y B

*Profesor:* César Fabián Domínguez Velasco

*Asignatura:* Fundamentos de Programación

*No. de práctica(s):* 10

*Integrante(s):* 13\_García\_Sánchez\_Alejandro  
18\_Lopez\_Castro\_Anastacia  
34\_Ramirez\_Rivas\_Gael  
39\_Ruiz\_Hernandez\_Ruben\_Antonio

*No. de lista o brigada:* 1A

*Semestre:* 2024-2

*Fecha de entrega:* 22 de Mayo del 2024

*Observaciones:*

**CALIFICACIÓN:** \_\_\_\_\_

## PRÁCTICA 10: ARREGLOS MULTIDIMENSIONALES

### 1) Objetivo:

El alumno utilizará arreglos de dos dimensiones en la elaboración de programas que resuelvan problemas que requieran agrupar datos del mismo tipo, en estructuras que utilizan dos índices.

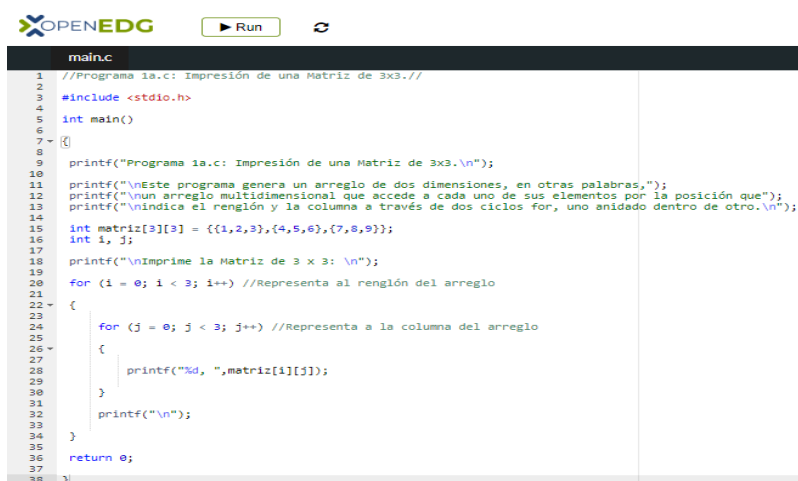
### 2) Introducción:

Un arreglo de dos dimensiones es un conjunto de datos contiguos del mismo tipo con un tamaño fijo definido al momento de crearse, de igual manera, para acceder a un elemento en este tipo de arreglos se requiere el uso de dos índices, por otro lado, se utilizan para hacer más eficiente el código de un programa, ya que la manipulación de datos del mismo tipo que son agrupados en un arreglo por tener un significado común se realiza de una forma más clara y eficaz.

A continuación, mostramos la construcción, compilación y ejecución de unos programas diseñados en lenguaje C, basados en esta segunda parte de arreglos.

### 3) Desarrollo (Capturas de pantalla de los programas en C):

#### → Programa 1a.c



```
1 //Programa 1a.c: Impresión de una Matriz de 3x3.//
2
3 #include <stdio.h>
4
5 int main()
6 {
7     printf("Programa 1a.c: Impresión de una Matriz de 3x3.\n");
8     printf("\nEste programa genera un arreglo de dos dimensiones, en otras palabras,");
9     printf("\nun arreglo multidimensional que accede a cada uno de sus elementos por la posición que");
10    printf("\nindica el renglón y la columna a través de dos ciclos for, uno anidado dentro de otro.\n");
11
12    int matriz[3][3] = {{1,2,3},{4,5,6},{7,8,9}};
13    int i, j;
14
15    printf("\nImprime la Matriz de 3 x 3: \n");
16    for (i = 0; i < 3; i++) //Representa al renglón del arreglo
17    {
18        for (j = 0; j < 3; j++) //Representa a la columna del arreglo
19        {
20            printf("%d, ",matriz[i][j]);
21        }
22        printf("\n");
23    }
24    return 0;
25 }
```

## Console

Programa 1a.c: Impresión de una Matriz de 3x3.

Este programa genera un arreglo de dos dimensiones, en otras palabras, un arreglo multidimensional que accede a cada uno de sus elementos por la posición que indica el renglón y la columna a través de dos ciclos for, uno anidado dentro de otro.

Imprime la Matriz de 3 x 3:

```
1, 2, 3,  
4, 5, 6,  
7, 8, 9,
```

## → Programa 2a.c



▶ Run



main.c

```
1 //Programa 2a.c: Impresión de una Matriz de 5x5.//  
2  
3 #include <stdio.h>  
4  
5 int main()  
6  
7 {  
8  
9     printf("Programa 2a.c: Impresión de una Matriz de 5x5.\n");  
10  
11     printf("\nEste programa genera un arreglo de dos dimensiones, en otras palabras,");  
12     printf("\nun arreglo multidimensional que accede a cada uno de sus elementos por la posición que");  
13     printf("\nindica el renglón y la columna a través de dos ciclos for, uno anidado dentro de otro,");  
14     printf("\nel contenido de cada elemento de este arreglo es la suma de sus índices.\n");  
15  
16     int i, j, a[5][5];  
17  
18     printf("\nImprime la Matriz de 5 x 5: \n");  
19  
20     for (i = 0; i < 5; i++) //Representa al renglón del arreglo  
21     {  
22  
23         for (j = 0; j < 5; j++) //Representa a la columna del arreglo  
24         {  
25  
26             {  
27  
28                 a[i][j] = i + j;  
29  
30                 printf("\t%d, ", a[i][j]);  
31  
32             }  
33  
34             printf("\n");  
35  
36         }  
37  
38     }  
39  
40     return 0;  
41 }
```

## Console

Programa 2a.c: Impresión de una Matriz de 5x5.

Este programa genera un arreglo de dos dimensiones, en otras palabras, un arreglo multidimensional que accede a cada uno de sus elementos por la posición que indica el renglón y la columna a través de dos ciclos for, uno anidado dentro de otro, el contenido de cada elemento de este arreglo es la suma de sus índices.

Imprime la Matriz de 5 x 5:

```
0,      1,      2,      3,      4,  
1,      2,      3,      4,      5,  
2,      3,      4,      5,      6,  
3,      4,      5,      6,      7,  
4,      5,      6,      7,      8,
```

## → Programa 1b.c

```
main.c
1 //Programa1b.c: Impresión de una matriz ocupando un ciclo while//
2
3 #include <stdio.h>
4
5 int main()
6 {
7
8
9     printf("Este programa permite imprimir una matriz de tamaño 3 x 3, ocupando un ciclo while\n");
10
11     int matriz[3][3] = {{1,2,3}, {4,5,6}, {7,8,9}};
12     int i, j;
13
14     printf("\nImprimir Matriz:\n");
15
16     i = 0;
17
18     while(i < 3) //Representa al renglón del arreglo
19     {
20         j = 0;
21
22         while (j < 3)//Representa a la columna del arreglo
23         {
24             printf("%d, ",matriz[i][j]);
25             j++;
26         }
27         printf("\n");
28         i++;
29     }
30
31     return 0;
32 }
```

### Console

Este programa permite imprimir una matriz de tamaño 3 x 3, ocupando un ciclo while

Imprimir Matriz:

```
1, 2, 3,
4, 5, 6,
7, 8, 9,
```

## → Programa 2b.c



▶ Run



```
main.c
1 //Programa 2b.c: Impresión de una Matriz de 5x5.//
2
3 #include <stdio.h>
4
5 int main()
6 {
7
8
9     printf("Programa 2b.c: Impresión de una Matriz de 5x5.\n");
10
11     printf("\nEste programa genera un arreglo de dos dimensiones, en otras palabras,");
12     printf("\nun arreglo multidimensional que accede a cada uno de sus elementos por la posición que");
13     printf("\nindica el renglón y la columna a través de dos ciclos while, uno anidado dentro de otro,");
14     printf("\nel contenido de cada elemento de este arreglo es la suma de sus índices.\n");
15
16     int i, j, a[5][5];
17     i = 0;
18
19     printf("\nImprime la Matriz de 5 x 5: \n");
20
21     while (i < 5) //Representa al renglón del arreglo
22     {
23         j = 0;
24
25         while (j<5) //Representa a la columna del arreglo
26         {
27             a[i][j] = i + j;
28             printf("\t%d, ",a[i][j]);
29             j++;
30         }
31         printf("\n");
32         i++;
33     }
34
35     return 0;
36 }
```

[Share Code](#)

ANSI C

### Console

Programa 2b.c: Impresión de una Matriz de 5x5.

Este programa genera un arreglo de dos dimensiones, en otras palabras, un arreglo multidimensional que accede a cada uno de sus elementos por la posición que indica el renglón y la columna a través de dos ciclos while, uno anidado dentro de otro, el contenido de cada elemento de este arreglo es la suma de sus índices.

Imprime la Matriz de 5 x 5:

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

### → Programa 1c.c

```
main.c
1  #include<stdio.h>
2
3  int main()
4  {
5      int matriz[3][3] = {{1,2,3},{4,5,6},{7,8,9}};
6      int i, j;
7      printf("Imprimir Matriz\n");
8      i=0;
9      do //Representa al renglón del arreglo
10     {
11         j=0;
12         do //Representa a la columna del arreglo
13         {
14             printf("%d, ",matriz[i][j]);
15             j++;
16         }
17
18         while (j<3);
19         printf("\n");
20         i++;
21     }
22
23     while(i<3);
24     return 0;
25 }
26
27 }
```

```
1, 2, 3,
4, 5, 6,
7, 8, 9,
```

```
...Program finished with exit code 0
Press ENTER to exit console.
```

→ Programa 2c.c

```
main.c
1  #include<stdio.h>
2
3  int main()
4  {
5      int i,j,a[5][5];
6      i=0;
7      do //Representa al renglón del arreglo
8      {
9          j=0;
10         do //Representa a la columna del arreglo
11         {
12             a[i][j]=i+j;
13             printf("\t%d", a[i][j]);
14             j++;
15         }
16         while (j<5);
17         printf("\n");
18         i++;
19     }
20     while (i<5);
21     return 0;
22 }
```

```

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

...Program finished with exit code 0
Press ENTER to exit console.
```

### → Programa 3.c

```
.dev] - Dev-C++ 5.11
View Project Execute Tools Style Window Help
bugals)
bug [*] main.cpp
1  #include <stdio.h>
2  int main ()
3  {
4  int lista[10][10]; // Se declara el arreglo multidimensional
5  int i,j;
6  int renglon,columna;
7  printf("\nDa el número de renglones y columnas separados con coma\n");
8  scanf("%d,%d",&renglon,&columna);
9  if(((renglon>=1) && (renglon<=10))&&((columna>=1) && (columna<=10)))
10 {
11 // Acceso a cada elemento del arreglo multidimensional usando for
12 for (i= 0 ; i <= renglon-1 ; i++)
13 {
14 for(j= 0 ; j <= columna-1 ; j++)
15 {
16 printf("\nNúmero para el elemento %d,%d del arreglo", i,j );
17 scanf("%d",&lista[i][j]);
18 }
19 }
20 printf("\nLos valores dados son: \n");
21 // Acceso a cada elemento del arreglo multidimensional usando for
22 for (i= 0 ; i <= renglon-1 ; i++)
23 {
24 for(j= 0 ; j <= columna-1 ; j++)
25 {
26 printf("%d ", lista[i][j]);
27 }
28 printf("\n");
29 }
30 }
31 else printf("Los valores dados no es válido");
32 printf("\n");
```

```
Da el número de renglones y columnas separados con coma
3,2
```

```
Número para el elemento 0,0 del arreglo4
```

```
Número para el elemento 0,1 del arreglo6
```

```
Número para el elemento 1,0 del arreglo7
```

```
Número para el elemento 1,1 del arreglo8
```

```
Número para el elemento 2,0 del arreglo9
```

```
Número para el elemento 2,1 del arreglo2
```

```
Los valores dados son:
```

```
4 6
```

```
7 8
```

```
9 2
```

```
-----
Process exited after 18.7 seconds with return value 0
Presione una tecla para continuar . . . |
```

## → Programa 4a.c

```
main.cpp
1  #include<stdio.h>
2  int main()
3  {
4      int matriz[3][3] = {{1,2,3},{4,5,6},{7,8,9}};
5      int i, cont=0, *ap;
6      ap = *matriz; //Esta sentencia es análoga a: ap = &matriz[0][0];
7      printf("Imprimir Matriz\n");
8      for (i=0 ; i<9 ; i++)
9      {
10         if (cont == 3) //Se imprimió un renglón y se hace un salto de línea
11         {
12             printf("\n");
13             cont = 0; //Inicia conteo de elementos del siguiente renglón
14         }
15         printf("%d\t",*(ap+i)); //Se imprime el siguiente elemento de la matriz
16         cont++;
17     }
18     printf("\n");
19     return 0;
20 }
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\anahe\OneDrive\Escritorio\Project1.exe
- Output Size: 128.7724609375 KiB
- Compilation Time: 2.77s

Imprimir Matriz

1	2	3
4	5	6
7	8	9

Process exited after 4.232 seconds with return value 0  
Presione una tecla para continuar . . .

## → Programa 4b.c

```
1 //El código del siguiente programa genera un arreglo de dos dimensiones (arreglo
2 //multidimensional) y accede a sus elementos a través de un apuntador utilizando
3 //un ciclo while.
4
5 #include<stdio.h>
6 int main()
7 {
8     int matriz[3][3] = {{1,2,3},{4,5,6},{7,8,9}};
9     int i, cont=0, *ap;
10    ap = *matriz;
11    printf("Imprimir Matriz\n");
12    i=0;
13    while (i<9)
14    {
15        if (cont == 3)
16        {
17            printf("\n");
18            cont = 0;
19        }
20        printf("%d\t",*(ap+i));
21        cont++;
22        i++;
23    }
24    printf("\n");
25    return 0;
26 }
```

Imprimir Matriz

1	2	3
4	5	6
7	8	9

...Program finished with exit code 0  
Press ENTER to exit console.



## → Programa 4c.c

```
1 //El código del siguiente programa genera un arreglo de dos dimensiones (arreglo
2 //multidimensional) y accede a sus elementos a través de un apuntador
3 //utilizando un ciclo do-while
4
5 #include<stdio.h>
6 int main()
7 {
8     int matriz[3][3] = {{1,2,3},{4,5,6},{7,8,9}};
9     int i, cont=0, *ap;
10    ap = &matriz[0][0]; //Esta sentencia es análoga a: ap = &matriz[0][0];
11    printf("Imprimir Matriz\n");
12    i=0;
13    do
14    {
15        if (cont == 3) //Se imprimió un renglón y se hace un salto de línea
16        {
17            printf("\n");
18            cont = 0; //Inicia conteo de elementos del siguiente renglón
19        }
20        printf("%d\t",*(ap+i)); //Se imprime el siguiente elemento de la matriz
21        cont++;
22        i++;
23    }
24    while (i<9);
25    printf("\n");
26    return 0;
27 }
```

Input

```
Imprimir Matriz
1      2      3
4      5      6
7      8      9

.. Program finished with exit code 0
Press ENTER to exit console
```

## 4) Conclusiones (Individuales):

- ❖ **García Sánchez Alejandro:** Los programas 4b.c y 4c.c fueron de gran utilidad para entender un poco más sobre arreglos multidimensionales y cómo implementar ciclos While y do-while para acceder a sus elementos, estos ejercicios fueron muy prácticos y de gran utilidad para poder implementarlos posteriormente en algún arreglo matricial.
- ❖ **López Castro Anastacia:** Este código en C declara y maneja un arreglo multidimensional de  $i \times j$ , permitiendo al usuario ingresar el número de filas y columnas, así como los valores para cada elemento del arreglo. Después de ingresar los datos, el programa los imprime en formato de matriz.  
  
En resumen, este programa permite al usuario definir las dimensiones de un arreglo 2D dentro de un rango específico, ingresar valores para cada posición del arreglo, y luego imprime esos valores en formato de matriz.
- ❖ **Ramírez Rivas Gael:** En esta práctica se revisaron programas basados en la elaboración y en algunos usos de los arreglos, principalmente usados para la elaboración de distintas matrices. El desarrollo y prueba

de los programas no presentó complicaciones, permitiéndonos desarrollar correctamente la práctica. El uso correcto de dichos arreglos es útil para desarrollar distintos objetivos o resolver problemas, por lo tanto, los conocimientos adquiridos al cumplir los objetivos en esta práctica son útiles.

- ❖ **Ruíz Hernández Rubén Antonio:** Se cumplieron al pie de la letra los objetivos ya que desde el programa 1a.c al 2b.c vimos la continuidad de arreglos donde ahora construimos e imprimimos nuestros primeros arreglos bidimensionales, donde se ejecutaron, pero sobretodo compilamos programas usando las estructuras de control iterativas (for y while) para imprimir matrices de tamaño 3x3 y 5x5.

#### **5) Retroalimentación (Equipo):**

Con base a los objetivos de la práctica tuvimos fallas al plantear la construcción de una matriz con respecto a la impresión de los mensajes, incluyendo el manejo de la estructura de control for pero pudimos lograr paso a paso el manejo de arreglos de dos dimensiones con base a la elaboración de programas que resolvieron problemas que requirieron agrupar datos del mismo tipo. En cambio, lo que nos ayudó durante la semana fueron programas que el asesor planteó sobre la construcción de una matriz, asimismo, nosotros modificamos el código de la clase del 13 de Mayo del 2024, donde abarcamos la programación de una matriz de 2 x 2 haciendo el uso de la estructura de control for para llevar a cabo la impresión de dicha matriz sin emplear una función, de igual manera logramos sacar el determinante de aquella, por lo cual gracias a este problema pudimos corregir al final las dificultades surgidas; del mismo modo empleamos el uso de las terminales Dev C++, Online Pascal y OPENEDG.

#### **6) Fuentes en APA:**

- ★ **Laboratorio Salas A y B. (s.f.). *Manual de Prácticas de la Asignatura Fundamentos de Programación (Guía práctica de estudio 10:***

*Arreglos Multidimensionales*, pág. 151 - 163). Recuperado el 19 de Abril del 2024, de Laboratorio de Computación Salas A y B:  
<http://lcp02.fi-b.unam.mx/>

- ★ RuyAntonio. (s.f.). *GitHub - RuyAntonio/practica10\_fdp: Práctica 10: Arreglos Multidimensionales*, de GitHub. Recuperado el 19 de Abril del 2024, de GitHub:  
[https://github.com/RuyAntonio/practica10\\_fdp.git](https://github.com/RuyAntonio/practica10_fdp.git)