

# Laboratorios de computación salas A y B

<i>Profesor:</i>	<i>Alejandro Esteban Pimentel Alarcón</i>
<i>Asignatura:</i>	<i>Fundamentos de Programación</i>
<i>Grupo:</i>	<i>3</i>
<i>No de Práctica(s):</i>	<i>11</i>
<i>Integrante(s):</i>	<i>Flores Rodríguez Ricardo Manuel Valencia Moya José Alejandro</i>
<i>No. de Equipo de cómputo empleado:</i>	<i>51 y 52</i>
<i>No. de Lista o Brigada:</i>	<i>9795 8543</i>
<i>Semestre:</i>	<i>2020-1</i>
<i>Fecha de entrega:</i>	<i>28-octubre-19</i>
<i>Observaciones:</i>	

CALIFICACIÓN: \_\_\_\_\_

## Introducción

En esta práctica vamos a ver la importancia y utilidad de los arreglos pero primero tenemos que saber que son. Los arreglos son estructuras de datos consistentes en un conjunto de datos del mismo tipo. Los arreglos tienen un tamaño que es la cantidad de objetos del mismo tipo que pueden almacenar. Los arreglos son entidades estáticas debido a que se declaran de un cierto tamaño y conservan éste todo a lo largo de la ejecución del programa en el cual fue declarado.

## Objetivos

Reconocer la importancia y utilidad de los arreglos en la elaboración de programas que resuelvan problemas que requieran agrupar datos del mismo tipo, así como trabajar con arreglos tanto unidimensionales como multidimensionales.

## Actividad 1

Hacer un programa que:

- Pida al usuario un número.
- Genere un arreglo de esa longitud.
- Pida al usuario números suficientes para llenar el arreglo.
- Muestre al usuario el número menor y el mayor de dicho arreglo.

```
Ac4.c
1  #include <stdio.h>
2  int main(){
3      int num;
4      printf("Pon un numero para generar el arreglo de esa longitud\n");
5      scanf("%i",&num);
6      int lista[num];
7
8      for (int i = 0; i < num; ++i)
9      {
10         printf("lista[%i]\n",i);
11         scanf("%i",&lista[i]);
12     }
13
14     int a;
15     a=lista[0];
16     for (int i = 1; i < num; ++i)
17     {
18         if (lista[i]<a)
19         {
20             a=lista[i];
21         }
22     }
23     int b;
24     b=lista[0];
25     for (int i = 1; i < num; ++i)
26     {
27         if (lista[i]>b)
28         {
29             b=lista[i];
30         }
31     }
32     printf("El numero menor es %i\n",a);
33     printf("El numero mayor es %i\n",b);
```

```
[iMac-de-mac:Desktop mac$ ./main
Pon un numero para generar el arreglo de esa longitud
9
lista[0]
4
lista[1]
6
lista[2]
8
lista[3]
2
lista[4]
2
lista[5]
3
lista[6]
4
lista[7]
6
lista[8]
8
El numero menor es 2
El numero mayor es 8
iMac-de-mac:Desktop mac$ █
```

---

## Actividad 2

Hacer un programa que:

- Pida al usuario un dos números N y M.
- Genere dos matrices de  $N \times M$ .
- Pida al usuario números suficientes para llenar ambas matrices.
- Muestre al usuario la matriz resultado de sumar las dos de entrada.

```
1  #include <stdio.h>
2  int main (){
3      int x, y ;
4      printf("Ingrese columnas de la matriz\n");
5      scanf("%i",&x);
6      printf("Ingrese filas de la matriz\n");
7      scanf("%i",&y);
8
9      int m1[x][y];
10     int m2[x][y];
11     int m3[x][y];
12     printf("Llenar la matriz 1\n");
13     printf("\n");
14     for (int a = 0; a < x; a++)
15     {
16         for (int b = 0; b < y; b++)
17         {
18             printf("lugar[%i][%i]\n",a+1,b+1);
19             scanf("%i",&m1[a][b]);
20         }
21     }
22     printf("\n");
23     printf("Llenar la matriz 2\n");
24     printf("\n");
25     for (int a = 0; a < x; a++)
26     {
27         for (int b = 0; b < y; b++)
28         {
29             printf("lugar[%i][%i]\n",a+1,b+1);
30             scanf("%i",&m2[a][b]);
31         }
32     }
33     printf("\n");
34     printf("La suma de las dos matrices es:\n");
35     for (int a = 0; a < x; a++)
36     {
37         for (int b = 0; b < y; b++)
38         {
39             m3[a][b]=m1[a][b]+m2[a][b];
40             printf("%i\t",m3[a][b]);
41         }
42         printf("\n");
43     }
44
45     return 0;
46 }
```

```
iMac-de-mac:Desktop mac$ ./main
Ingrese columnas de la matriz
5
Ingrese filas de la matriz
2
Llenar la matriz 1

lugar[1][1]
2
lugar[1][2]
2
lugar[2][1]
3
lugar[2][2]
4
lugar[3][1]
5
lugar[3][2]
6
lugar[4][1]
7
lugar[4][2]
8
lugar[5][1]
5
lugar[5][2]
6

Llenar la matriz 2

lugar[1][1]
5
lugar[1][2]
7
lugar[2][1]
8
lugar[2][2]
8
lugar[3][1]
9
lugar[3][2]
7
lugar[4][1]
8
lugar[4][2]
5
lugar[5][1]
6
lugar[5][2]
8

La suma de las dos matrices es:
7      9
11     12
14     13
15     13
11     14
iMac-de-mac:Desktop mac$ █
```

## Conclusiones

Los arreglos son fundamentales en esta práctica por que primero tenemos que saber que son para poder llevarla a cabo, tienen una función muy importante para la elaboración de programas con matrices, no son muy difíciles de usar pero tenemos que razonar bien para no equivocarnos, y si se pudieron cumplir los objetivos a la perfección.