



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Marco Antonio Martínez Quintana

Profesor:

Fundamentos de programación

Asignatura:

3

Grupo:

11

No de Práctica(s):

Zurita León Dana Cecilia

Integrante(s):

*No. de Equipo de
cómputo empleado:*

No aplica

No. de Lista o Brigada:

Primero

Semestre:

04 de Enero de 2021

Fecha de entrega:

Observaciones:

CALIFICACIÓN: _____

Arreglos unidimensionales y multidimensionales

Objetivo:

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.

Introducción:

Los arreglos son conjuntos de datos del mismo tipo, a estos valores se les asigna una posición determinada, iniciando desde el número 0, tienen un tamaño fijo, este se determina como en la parte de asignaciones.

Los arreglos son de tipo multidimensional o unidimensional.

Los apuntadores son variables que contiene la dirección de una variable, esto se refiere a la localidad de memoria de otra variable, ya que trabaja directamente con la memoria, se obtienen datos de manera rápida.

Solo pueden acceder a datos con el mismo tipo de datos con el que fueron declarados.

Actividad:

```
1 #include <stdio.h>
2
3 /*
4  Este programa genera un arreglo unidimensional de 5 elementos y los
5  accede a cada elemento del arreglo a través de un ciclo while.
6  */
7 int main () {
8     //Define el tamaño del arreglo
9     #define TAMANO 5
10    //Determinacion de variables
11    int lista[TAMANO] = {10, 8, 5, 8, 7};
12    char ao= 'a';
13    int indice = 0;
14
15    //Imprime el nombre el programa
16    printf("\tLista\n");
17    //Se realiza el procedimiento siempre y cuando se cumpla la condicion
18    while (indice < 5 ) {
19        //Codigo a ejecutar
20        printf("\nCalificaci%cn del alumno %d es %d",ao, indice+1, lista[indice]);
21        indice += 1; // análogo a indice = indice + 1;
22    }
23
24    printf("\n");
25    return 0;
26 }
```

C:\> Símbolo del sistema

C:\Users\dellR\Desktop\LenguajeC\Ejemplos>gcc prac11.c -o prac11.exe

C:\Users\dellR\Desktop\LenguajeC\Ejemplos>prac11.exe
Lista

Calificación del alumno 1 es 10

Calificación del alumno 2 es 8

Calificación del alumno 3 es 5

Calificación del alumno 4 es 8

Calificación del alumno 5 es 7

C:\Users\dellR\Desktop\LenguajeC\Ejemplos>

```
fagausswhie.c x submenu.c x ifelsecalcu.c x menusalir.c x wcalculadora.c x p10.c x pp10.c x dep1.c x prac11.c x prac112.c x
1  #include <stdio.h>
2  /*
3   * Este programa genera un arreglo unidimensional de 5 elementos y
4   * accede a cada elemento del arreglo a través de un ciclo for.
5   */
6  int main () {
7      //Tamaño del arreglo
8      #define TAMANO 5
9      //Declaracion de variables
10     int lista[TAMANO] = {10, 8, 5, 8, 7};
11     char ao= 'a';
12     //Impresion de datos
13     printf("\tLista\n");
14     for (int indice = 0 ; indice < 5 ; indice++){
15         printf("\nCalificaci%cn del alumno %d es %d",ao, indice+1, lista[indice]);
16     }
17
18     //Salto de linea
19     printf("\n");
20
21     return 0;
22 }
```

C:\. Símbolo del sistema

```
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>gcc prac112.c -o prac112.exe
```

```
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>prac112.exe
Lista
```

```
Calificación del alumno 1 es 10
Calificación del alumno 2 es 8
Calificación del alumno 3 es 5
Calificación del alumno 4 es 8
Calificación del alumno 5 es 7
```

```
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>_
```

```
c11.c | prac112.c | prac113.c |
#include <stdio.h>
/*
 * Este programa crea un apuntador de tipo carácter.
 */
int main () {
    // Declaración del apuntador
    char *ap, c = 'a';
    ap = &c;
    // Declaración de accents
    char ao= 162, aa=160;

    //Impresión de datos en pantalla
    //Imprime el valor de la variable que se le asigno al apuntador
    printf("Carácter: %c\n",aa,*ap);
    //Imprime el valor de la variable que se asigno al apuntador en código ASCII
    printf("Código ASCII: %d\n",ao,*ap);
    //Imprime el valor decimal de la variable del apuntador
    printf("Dirección de memoria: %d\n",ao,ap);

    return 0;
}
```

Estándar para el Intercambio de Información)

CA. Símbolo del sistema

```
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>gcc prac113.c -o prac113.exe
```

```
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>prac113.exe
```

```
Carácter: a
```

```
Código ASCII: 97
```

```
Dirección de memoria: 6422295
```

```
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>_
```

```
pract11.c x pract112.c x pract113.c x pract114.c x
1  #include<stdio.h>
2  /*
3   * Este programa accede a las localidades de memoria de distintas variables a
4   * través de un apuntador.
5   */
6  int main () {
7      //Declaración de variables
8      int a = 5, b = 10, c[10] = {5, 4, 3, 2, 1, 9, 8, 7, 6, 0};
9      //Apuntador
10     int *apEnt;
11     apEnt = &a;
12     //Impresion en pantalla
13     //Indica los valores del arreglo
14     printf("a = 5, b = 10, c[10] = {5, 4, 3, 2, 1, 9, 8, 7, 6, 0}\n");
15     printf("apEnt = %a\n");
16
17     //Asigna que b es igual al valor asignado al apuntador
18     b = *apEnt;
19     printf("b = *apEnt \t-> b = %i\n", b);
20
21     //Asigna que b es igual al valor asignado al apuntador más uno
22     b = *apEnt + 1;
23     printf("b = *apEnt + 1 \t-> b = %i\n", b);
24
25     //Asigna que ahora el apuntador es igual a 0
26     *apEnt = 0;
27     printf("*apEnt = 0 \t-> a = %i\n", a);
28
29     //Asigna que apEnt ahora es igual al valor del arreglo en la posición 0
30     apEnt = &c[0];
31     printf("apEnt = &c[0] \t-> apEnt = %i\n", *apEnt);
32
33     return 0;
34 }
```

C:\ Símbolo del sistema

```
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>gcc prac114.c -o prac114.exe
```

```
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>prac114.exe
```

```
a = 5, b = 10, c[10] = {5, 4, 3, 2, 1, 9, 8, 7, 6, 0}
```

```
apEnt = &a
```

```
b = *apEnt      -> b = 5
```

```
b = *apEnt + 1  -> b = 6
```

```
*apEnt = 0      -> a = 0
```

```
apEnt = &c[0]    -> apEnt = 5
```

```
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>
```

```
#include <stdio.h>

/*
 * Este programa trabaja con aritmética de punteros para acceder a los
 * valores de un arreglo.
 */

int main () {
    //Declaración del arreglo
    int arr[] = {5, 4, 3, 2, 1};
    int *apArr;
    apArr = arr;

    //Impresión en pantalla
    //Imprime valores del arreglo
    printf("int arr[] = {5, 4, 3, 2, 1};\n");
    printf("apArr = &arr[0]\n");

    //La variable x es igual al apuntador
    int x = *apArr;
    //Imprime primer valor del arreglo
    printf("x = *apArr \t -> x = %d\n", x);

    //El apuntador toma el valor del segundo valor del arreglo ya que se le suma 1
    x = *(apArr+1);
    printf("x = *(apArr+1) \t -> x = %d\n", x);

    //El apuntador toma el tercer valor del arreglo
    x = *(apArr+2);
    printf("x = *(apArr+1) \t -> x = %d\n", x);

    return 0;
}
```


C:\> Símbolo del sistema

```
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>gcc prac115.c -o prac115.exe
```

```
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>prac115.exe
```

```
int arr[] = {5, 4, 3, 2, 1};
apArr = &arr[0]
x = *apArr      -> x = 5
x = *(apArr+1)  -> x = 4
x = *(apArr+1)  -> x = 3
```

```
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>
```

```
#include <stdio.h>

/*
 * Este programa genera un arreglo unidimensional de 5 elementos y
 * accede a cada elemento del arreglo a través de un apuntador
 * utilizando un ciclo for.
 */

int main () {
    //Declaración de variables
    char ao= 162;
    //Tamaño dle arreglo
    #define TAMANO 5
    //Valores del arreglo
    int lista[TAMANO] = {10, 8, 5, 8, 7};
    //Asignación del apuntador
    int *ap = lista;
    //Imprime título del programa
    printf("\tLista\n");
    //Ciclo que asigna a cada alumno su calificación segun el valor del arreglo correspondiente
    for (int indice = 0 ; indice < 5 ; indice++){
        //A cada apuntador se le suma uno
        printf("\nCalificación del alumno %d es %d",ao, indice+1, *(ap+indice));
    }
    //Salto de línea
    printf("\n");

    return 0;
}
```

C:\> Símbolo del sistema

```
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>gcc prac116.c -o prac116.exe
```

```
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>prac116.exe
Lista
```

```
Calificación del alumno 1 es 10
Calificación del alumno 2 es 8
Calificación del alumno 3 es 5
Calificación del alumno 4 es 8
Calificación del alumno 5 es 7
```

```
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>
```



```

#include <stdio.h>
/*
Este programa muestra el manejo de cadenas en lenguaje C.
*/
int main(){
    //Declaración de variables
    char palabra[20];
    int i=0;
    //Solicitud de datos
    printf("Ingrese una palabra: ");
    //Guarda la palabra escrita en %s
    scanf("%s", palabra);
    //Muestra la palabra con salto de línea en cada letra
    printf("La palabra ingresada es: %s\n", palabra);
    for (i = 0 ; i < 20 ; i++){
        printf("%c\n", palabra[i]);
    }
    return 0;
}

```

C:\ Símbolo del sistema

C:\Users\dellR\Desktop\LenguajeC\Ejemplos>>gcc prac117.c -o prac117.exe

C:\Users\dellR\Desktop\LenguajeC\Ejemplos>prac117.exe

Ingrese una palabra: Ceci

La palabra ingresada es: Ceci

C
e
c
i

↓
@

P
↓
@

@
,

C:\Users\dellR\Desktop\LenguajeC\Ejemplos>_

```
#include<stdio.h>

/* Este programa genera un arreglo de dos dimensiones (arreglo
multidimensional) y accede a sus elementos a través de dos ciclos
for, uno anidado dentro de otro.
*/

int main() {
    //Declaración de una matriz
    int matriz[3][3] = {{1,2,3},{4,5,6},{7,8,9}};
    //Declaración de variables
    int i, j;
    printf("Imprimir Matriz\n");
    //Imprime valores de la matriz
    for (i=0 ; i<3 ; i++){
        for (j=0 ; j<3 ; j++){
            printf("%d, ",matriz[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```

norm  Símbolo del sistema

```
1 C:\Users\dellR\Desktop\LenguajeC\Ejemplos>gcc prac118.c -o prac118.exe

C:\Users\dellR\Desktop\LenguajeC\Ejemplos>prac118.exe
Imprimir Matriz
1, 2, 3,
4, 5, 6,
7, 8, 9,

C:\Users\dellR\Desktop\LenguajeC\Ejemplos>
```

```
1 #include<stdio.h>
2 /* Este programa genera un arreglo de dos dimensiones (arreglo
3 multidimensional) y accede a sus elementos a través de un apuntador utilizando
4 un ciclo for.
5 */
6 int main(){
7     //Declaración de la matriz y sus valores
8     int matriz[3][3] = {{1,2,3},{4,5,6},{7,8,9}};
9     int i, cont=0, *ap;
10    ap = matriz;
11    //Impresion de datos en pantalla
12    printf("Imprimir Matriz\n");
13    //Se establece un contador, va desde cero hasta un número menor a nueve, agregando uno
14    for (i=0 ; i<9 ; i++){
15        //Cada tres valores se establece un salto de línea
16        if (cont == 3){
17            printf("\n");
18            cont = 0;
19        }
20        printf("%d\t",*(ap+i));
21        cont++;
22    }
23    printf("\n");
24    return 0;
25 }
```

```
100% Símbolo del sistema
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>gcc prac119.c -o prac119.exe
prac119.c: In function 'main':
prac119.c:9:7: warning: assignment to 'int *' from incompatible pointer type 'int (*)[3]' [-Wincompatible-pointer-types]
   9 |     ap = matriz;
     |         ^
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>prac119.exe
Imprimir Matriz
1      2      3
4      5      6
7      8      9
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>
```

```

1  #include<stdio.h>
2  int main ()
3  {
4      int ab[3][3];
5
6      printf("\n Abarrotes \n1) Dulceria\n2) Salchichoneria\n3) Lacteos\n");
7      printf("\n1) Dulces \n1.- Sandibrochas\n2.- Chocolates\n3.- Frituras V\n\n");
8      printf("\n2) Salchichoneria \n1.- Jamon\n2.- Queso\n3.- Salchichas\n\n");
9      printf("\n3) Lacteos \n1.- Yogur\n2.- Leche\n3.- Lechera\n\n");
10
11     for(int i=0; i<3; i++)
12     {
13         for(int j=0; j<3; j++)
14         {
15             printf("Numero de productos que desea %d de la seccion %d \t", j+1,162, i+1);
16             scanf("%d", &ab[i][j]);
17         }
18     }
19     /*printf("\nProductos totales\n",163);
20     for(int i=0; i<3; i++)
21     {
22         for(int j=0; j<3; j++)
23         {
24             printf("%d\t",ab[i][j]);
25         }
26         printf("\n");
27     }
28     printf("\n");*/
29
30     int p[3];
31     int a=0;
32     for(int i=0; i<3; i++)
33     {
34         a=0;
35         for(int j=0; j<3; j++)
36         {
37             a=a+ab[i][j];
38         }
39         p[i]=a;
40         printf("\nEl n°cmero total de productos del apartado %d son: %d", 163, i+1, p[i]);
41     }
42     a=0;
43     for(int i=0; i<3; i++)
44     {
45         a=a+p[i];
46     }
47     printf("\n\nEl n°cmero total de productos que desea son: %d\n", 163, a);
48
49     return 0;
50 }

```

CA% Símbolo del sistema

```
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>gcc p11.c -o p11.exe
```

```
C:\Users\dellR\Desktop\LenguajeC\Ejemplos>p11.exe
```

Abarrotes

- 1) Dulceria
- 2) Salchichoneria
- 3) Lacteos

- 1) Dulces
 - 1.- Sandibrochas
 - 2.- Chocolates
 - 3.- Frituras V

- 2) Salchichoneria
 - 1.- Jamon
 - 2.- Queso
 - 3.- Salchichas

- 3) Lacteos
 - 1.- Yogur
 - 2.- Leche
 - 3.- Lechera

Numero de productos que desea 1 de la sección 1	1
Numero de productos que desea 2 de la sección 1	0
Numero de productos que desea 3 de la sección 1	12
Numero de productos que desea 1 de la sección 2	67
Numero de productos que desea 2 de la sección 2	4
Numero de productos que desea 3 de la sección 2	10
Numero de productos que desea 1 de la sección 3	8
Numero de productos que desea 2 de la sección 3	5
Numero de productos que desea 3 de la sección 3	32

El número total de productos del apartado 1 son: 13
El número total de productos del apartado 2 son: 81
El número total de productos del apartado 3 son: 45

El número total de productos que desea son: 139

Conclusión:

En esta práctica pude aprender la importancia de los arreglos, estos hacen el código eficiente, también pude identificar como se determina un arreglo unidimensional y multidimensional.

No obstante también pude aprender como utilizar un apuntador, su sintaxis y la utilidad que tienen.

Fue una práctica con mucha retroalimentación.

Bibliografía

Guía práctica de estudio 11: Arreglos unidimensionales y multidimensionales
[PDF] Universidad Nacional Autónoma de México. Consultado en enero de
2021 del sitio: file:///C:/Users/dellR/Downloads/fp_p11.pdf