



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: Marco Antonio Martínez Quintana

Asignatura: Fundamentos de Programación

Grupo: 1103

No de Práctica(s): 11

Integrante(s): Ulises Castro Rodríguez

*No. de Equipo de
cómputo empleado:* No aplica

No. de Lista o Brigada: 08

Semestre: Primer Semestre

Fecha de entrega: Domingo 10/01/2021

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**

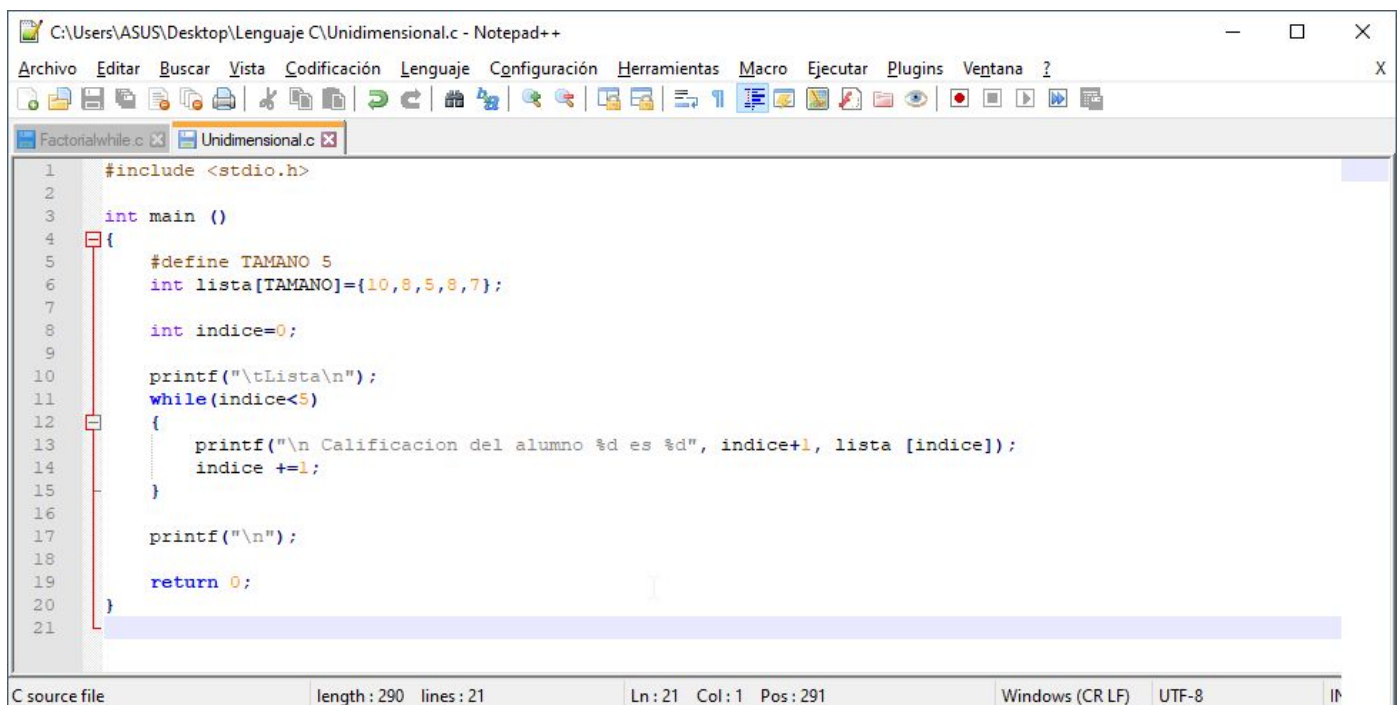
Un arreglo es un conjunto de datos contiguos del mismo tipo con un tamaño fijo definido al momento de crearse.

A cada elemento (dato) del arreglo se le asocia una posición particular, el cual se requiere indicar para acceder a un elemento en específico. Esto se logra a través del uso de índices. Los arreglos pueden ser unidimensionales o multidimensionales. Los arreglos se utilizan para hacer más eficiente el código de un programa.

- **Actividades**

1. Elaborar un programa en lenguaje C que emplee arreglos de una dimensión.

Código (arreglo unidimensional while)



```
1  #include <stdio.h>
2
3  int main ()
4  {
5      #define TAMANO 5
6      int lista[TAMANO]={10,8,5,8,7};
7
8      int indice=0;
9
10     printf("\tLista\n");
11     while(indice<5)
12     {
13         printf("\n Calificacion del alumno %d es %d", indice+1, lista [indice]);
14         indice +=1;
15     }
16
17     printf("\n");
18
19     return 0;
20 }
21
```

C source file length : 290 lines : 21 Ln : 21 Col : 1 Pos : 291 Windows (CR LF) UTF-8 IT

```
Símbolo del sistema

C:\Users\ASUS\Desktop\Lenguaje C>gcc Unidimensional.c -o Unidimensional.exe

C:\Users\ASUS\Desktop\Lenguaje C>Unidimensional.exe
"Unidimensional.exe" no se reconoce como un comando interno o externo,
programa o archivo por lotes ejecutable.

C:\Users\ASUS\Desktop\Lenguaje C>Unidimensional.exe
Lista

Calificacion del alumno 1 es 10
Calificacion del alumno 2 es 8
Calificacion del alumno 3 es 5
Calificacion del alumno 4 es 8
Calificacion del alumno 5 es 7

C:\Users\ASUS\Desktop\Lenguaje C>
```

2. Manipular arreglos a través de índices y apuntadores.

Codigo (apuntadores)

```
C:\Users\ASUS\Desktop\Lenguaje C\Apuntador.c - Notepad++

Archivo  Editar

#include <stdio.h>

int main ()
{
    char *ap, c='a';
    ap=&c;

    printf("Caracter: %c\n", *ap);
    printf("Codigo ASCII: %d\n", *ap);
    printf("Direccion de memoria: %d\n", ap);

    return 0;
}
```

C source file length : 199 lines : 13 Ln : 11 Col : 5 Pos : 185 Windows (CR LF) UTF-8 IP

```
Símbolo del sistema

C:\Users\ASUS\Desktop\Lenguaje C>gcc Apuntador.c -o Apuntador.exe

C:\Users\ASUS\Desktop\Lenguaje C>Apuntador.exe
Caracter: a
Codigo ASCII: 97
Direccion de memoria: 6422299

C:\Users\ASUS\Desktop\Lenguaje C>
```

Código (apuntadores en ciclo for)

```
C:\Users\ASUS\Desktop\Lenguaje C\Apuntador2.c - Notepad++
Archivo Editar Buscar Vista Codificación Lenguaje Configuración Herramientas Macro Ejecutar Plugins Ventana ?
Factorialwhile.c Unidimensional.c Apuntador2.c
1 #include <stdio.h>
2 //Este programa accede a las localidades de memoria de distintas variables
3 //a traves de un apuntador
4
5 int main ()
6 {
7     int a=5, b=10, c[10]={5,4,3,2,1,9,8,7,6,0};
8     int *apEnt;
9     apEnt=&a;
10
11     printf("a=5, b=10, c[10]={5,4,3,2,1,9,8,7,6,0}\n");
12     printf("apEnt=&a\n");
13
14     b=*apEnt;
15     printf("b=*apEnt \t->b=%i\n",b);
16
17     b=*apEnt+1;
18     printf("b=*apEnt+1 \t->b=%i\n",b);
19
20     *apEnt=0;
21     printf("*apEnt=0 \t->a=%i\n",a);
22
23     apEnt=&c[0];
24     printf("apEnt=&c[0] \t->apEnt=%i\n",*apEnt);
25
26     return 0;
27 }
```

C source file length: 534 lines: 27 Ln: 27 Col: 2 Pos: 535 Windows (CR LF) UTF-8 INS

```
Ca\ Símbolo del sistema
C:\Users\ASUS\Desktop\Lenguaje C>gcc Apuntador2.c -o Apuntador.exe
C:\Users\ASUS\Desktop\Lenguaje C>Apuntador.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\ASUS\Desktop\Lenguaje C>
```

Código (apuntadores en ciclo for)

```
C:\Users\ASUS\Desktop\Lenguaje C\Apuntador3.c - Notepad++
Archivo Editar Buscar Vista Codificación Lenguaje Configuración Herramientas Macro Ejecutar Plugins Ventana ?
Unidimensional.c Apuntador3.c
1 #include <stdio.h>
2
3 int main ()
4 {
5     #define TAMANO 5
6     int lista[TAMANO]={10,8,5,8,7};
7     int *ap=lista;
8
9     printf("\tLista\n");
10    for(int indice=0; indice<5; indice++)
11    {
12        printf("\nCalificacion del alumno %d es de %d", indice+1, *(ap+indice));
13    }
14    return 0;
15 }
```

```
Símbolo del sistema

C:\Users\ASUS\Desktop\Lenguaje C>gcc Apuntador3.c -o Apuntador.exe

C:\Users\ASUS\Desktop\Lenguaje C>Apuntador.exe
Lista

Calificacion del alumno 1 es de 10
Calificacion del alumno 2 es de 8
Calificacion del alumno 3 es de 5
Calificacion del alumno 4 es de 8
Calificacion del alumno 5 es de 7
C:\Users\ASUS\Desktop\Lenguaje C>
```

Código (apuntadores en cadenas)

```
C:\Users\ASUS\Desktop\Lenguaje C\Apuntador4.c - Notepad++
Archivo Editar Buscar Vista Codificación Lenguaje Configuración Herramientas Macro Ejecutar
Unidimensional.c Apuntador3.c Apuntador4.c
1 #include <stdio.h>
2
3 int main ()
4 {
5     char palabra[20];
6     int i=0;
7
8     printf("Ingrese una palabra: ");
9     scanf("%s", palabra);
10    printf("La palabra ingresada es: %s\n", palabra);
11
12    for (i=0; i<20; i++)
13    {
14        printf("%c\n", palabra[i]);
15    }
16    return 0;
17 }
```

```
Símbolo del sistema

C:\Users\ASUS\Desktop\Lenguaje C>gcc Apuntador4.c -o Apuntador.exe

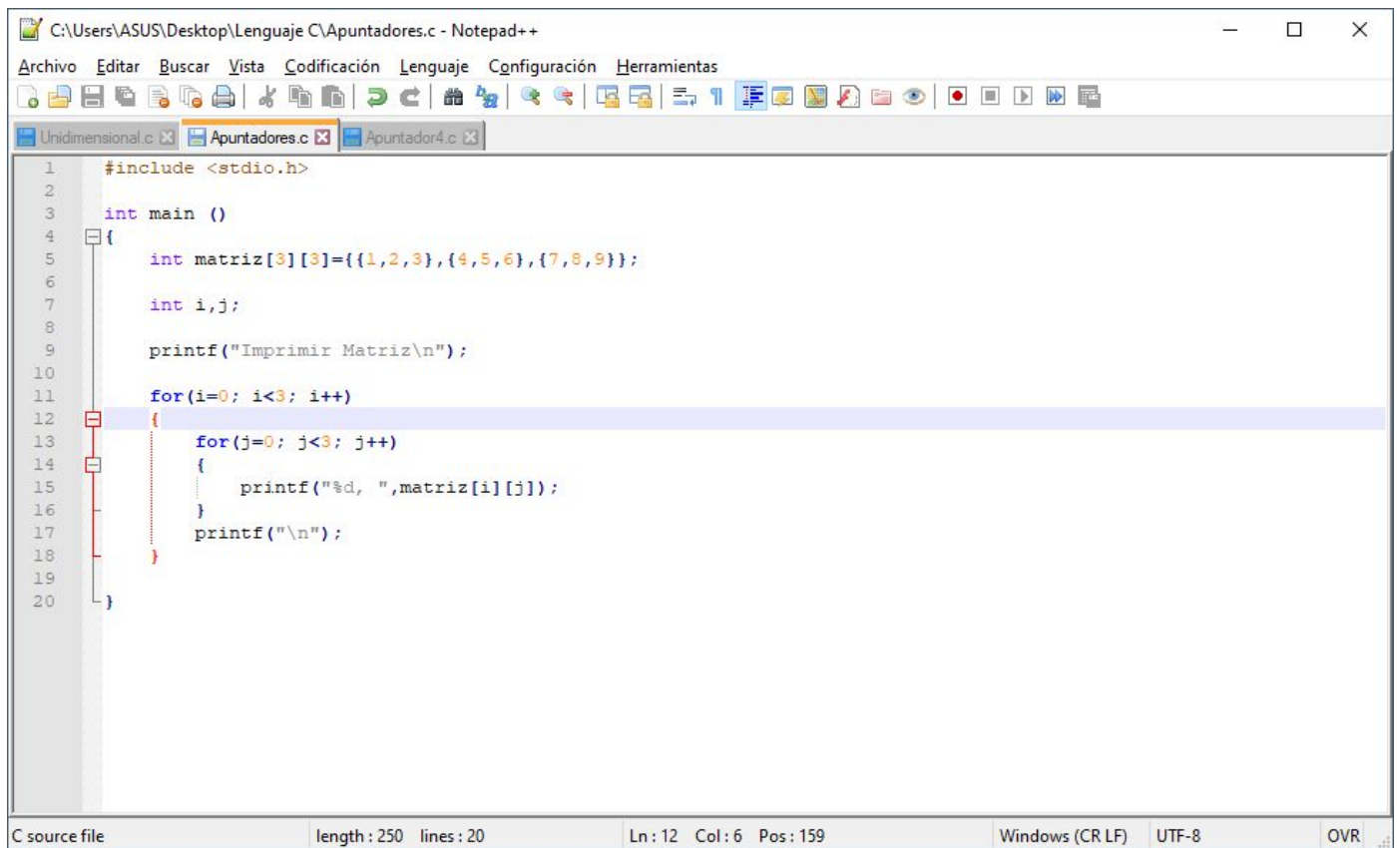
C:\Users\ASUS\Desktop\Lenguaje C>Apuntador.exe
Ingrese una palabra: Ulises
La palabra ingresada es: Ulises
U
l
i
s
e
s

p
↓
@

p
↓

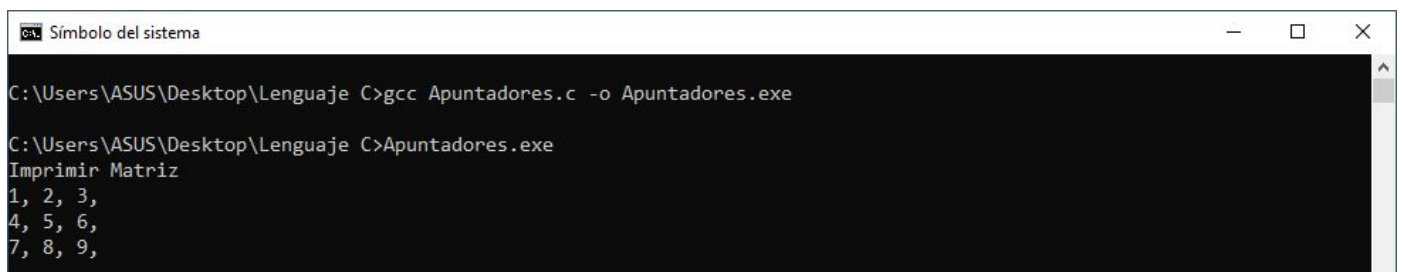
C:\Users\ASUS\Desktop\Lenguaje C>
```


Código (arreglos multidimensionales)



```
1  #include <stdio.h>
2
3  int main ()
4  {
5      int matriz[3][3]={{1,2,3},{4,5,6},{7,8,9}};
6
7      int i,j;
8
9      printf("Imprimir Matriz\n");
10
11     for(i=0; i<3; i++)
12     {
13         for(j=0; j<3; j++)
14         {
15             printf("%d, ",matriz[i][j]);
16         }
17         printf("\n");
18     }
19
20 }
```

C source file length : 250 lines : 20 Ln : 12 Col : 6 Pos : 159 Windows (CR LF) UTF-8 OVR



```
Simbolo del sistema
C:\Users\ASUS\Desktop\Lenguaje C>gcc Apuntadores.c -o Apuntadores.exe
C:\Users\ASUS\Desktop\Lenguaje C>Apuntadores.exe
Imprimir Matriz
1, 2, 3,
4, 5, 6,
7, 8, 9,
```

Conclusiones

El día de hoy aprendí acerca de los arreglos utilizados en la programación en lenguaje c que son conjuntos de datos almacenados que tienen un valor definido, como pudimos ver pueden ser unidimensionales o multidimensionales (matrices), este recurso es muy útil a la hora de crear programas ya que nos ayuda en la optimización de un código, como se nos explicó en la práctica, Además en esta práctica conocimos los apuntadores que son herramientas que nos permiten almacenar datos en la memoria para después hacer uso de ellos en el código. Esta práctica me pareció entretenida e interesante, aunque tuve un pequeño problema con el último ejercicio ya que no me permitía compilarlo.

• Bibliografías.

- El lenguaje de programación C. Brian W. Kernighan, Dennis M. Ritchie, segunda edición, USA, Pearson Educación 1991.