	Manual de prácticas del Laboratorio de Estructuras de datos y algoritmos I	Código:	MADO-19
		Versión:	01
		Página	3/151
		Sección ISO	8.3
		Fecha de emisión	20 de enero de 2017
Facultad de Ingeniería		Área/Departamento: Laboratorio de computación salas A y B	
La impresión de este documento es una copia no controlada			

Guía práctica de estudio 01: Aplicaciones de arreglos




Elaborado por:

M.C. Edgar E. García Cano
Ing. Jorge A. Solano Gálvez

Autorizado por:

M.C. Alejandro Velázquez Mena

	Manual de prácticas del Laboratorio de Estructuras de datos y algoritmos I	Código:	MADO-19
		Versión:	01
		Página	4/151
		Sección ISO	8.3
		Fecha de emisión	20 de enero de 2017
Facultad de Ingeniería		Área/Departamento: Laboratorio de computación salas A y B	
La impresión de este documento es una copia no controlada			

Guía práctica de estudio 01: Aplicaciones de arreglos

Objetivo:

Utilizar arreglos unidimensionales y multidimensionales para dar solución a problemas computacionales.

Actividades:

- Crear arreglos unidimensionales.
- Crear arreglos multidimensionales.

Introducción

Un arreglo es un conjunto de datos contiguos del mismo tipo con un tamaño fijo definido al momento de crearse. Pueden ser unidimensionales o multidimensionales.

A cada elemento (dato) del arreglo se le asocia una posición particular. Para acceder a los elementos de un arreglo es necesario utilizar un índice. En lenguaje C, el índice de cada dimensión inicia en 0 y termina en $n-1$, donde n es el tamaño de la dimensión.


Licencia GPL de GNU

El software presente en esta guía práctica es libre bajo la licencia GPL de GNU, es decir, se puede modificar y distribuir mientras se mantenga la licencia GPL.

```

/*
 * This program is free software: you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation, either version 3 of the License, or
 * (at your option) any later version.
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the

```

	Manual de prácticas del Laboratorio de Estructuras de datos y algoritmos I	Código:	MADO-19
		Versión:	01
		Página	5/151
		Sección ISO	8.3
		Fecha de emisión	20 de enero de 2017
Facultad de Ingeniería		Área/Departamento: Laboratorio de computación salas A y B	
La impresión de este documento es una copia no controlada			

```

* GNU General Public License for more details.
*
* You should have received a copy of the GNU General Public License
* along with this program. If not, see <http://www.gnu.org/licenses/>.
*
* Author: Jorge A. Solano
*/

```

Arreglos contiguos o ligados

Un arreglo contiguo es aquel que se crea desde el inicio del programa y permanece estático durante toda la ejecución del mismo, es decir, no se puede redimensionar.

Un arreglo ligado es aquel que se declara en tiempo de ejecución y bajo demanda, por lo tanto, es posible incrementar su tamaño durante la ejecución del programa, utilizando de manera más eficiente la memoria. Para crear un arreglo ligado se debe utilizar lo que se conoce como memoria dinámica.

Los arreglos unidimensionales están constituidos por localidades de memoria (ya sea contiguas o ligadas) ordenadas bajo un mismo nombre y sobre un solo nivel (una dimensión).

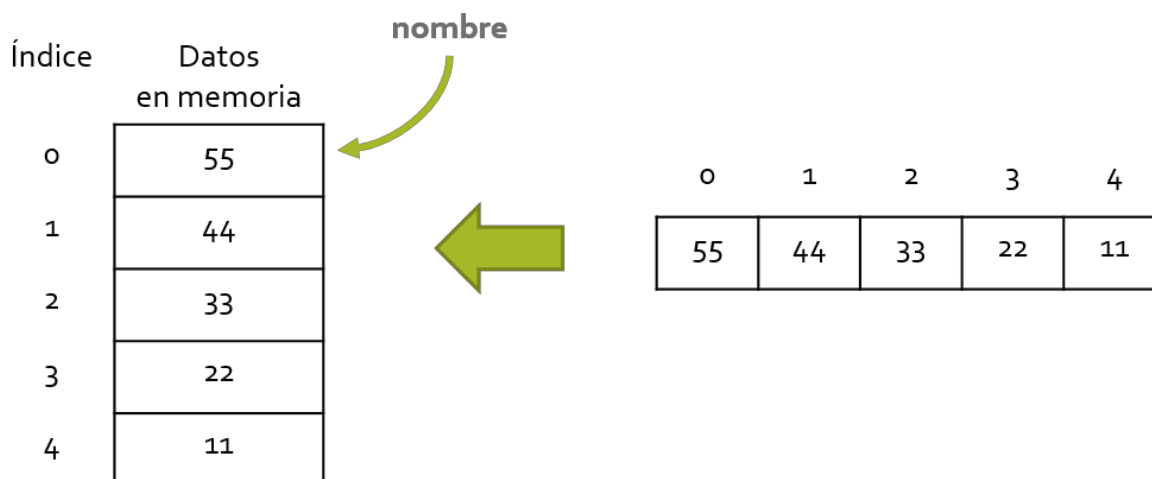



Figura 1. Representación en memoria de un arreglo unidimensional.

	Manual de prácticas del Laboratorio de Estructuras de datos y algoritmos I	Código:	MADO-19
		Versión:	01
		Página	6/151
		Sección ISO	8.3
		Fecha de emisión	20 de enero de 2017
Facultad de Ingeniería		Área/Departamento: Laboratorio de computación salas A y B	
La impresión de este documento es una copia no controlada			

Los arreglos multidimensionales están constituidos por localidades de memoria (ya sea contiguas o ligadas) ordenadas bajo un mismo nombre y que pueden tener varios niveles (varias dimensiones) que van desde el plano (2 dimensiones) hasta la enésima dimensión.

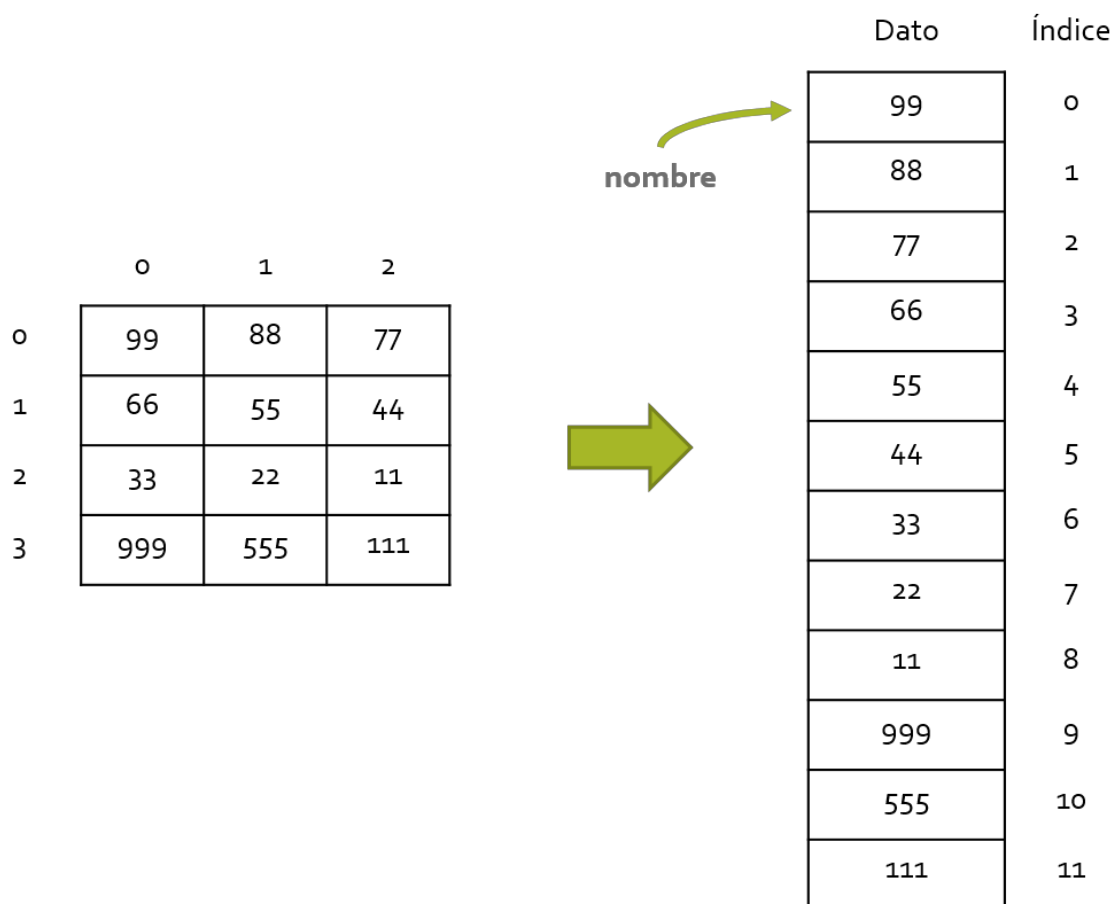



Figura 2. Representación en memoria de un arreglo bidimensional.

	Manual de prácticas del Laboratorio de Estructuras de datos y algoritmos I	Código:	MADO-19
		Versión:	01
		Página	7/151
		Sección ISO	8.3
		Fecha de emisión	20 de enero de 2017
Facultad de Ingeniería		Área/Departamento: Laboratorio de computación salas A y B	
La impresión de este documento es una copia no controlada			

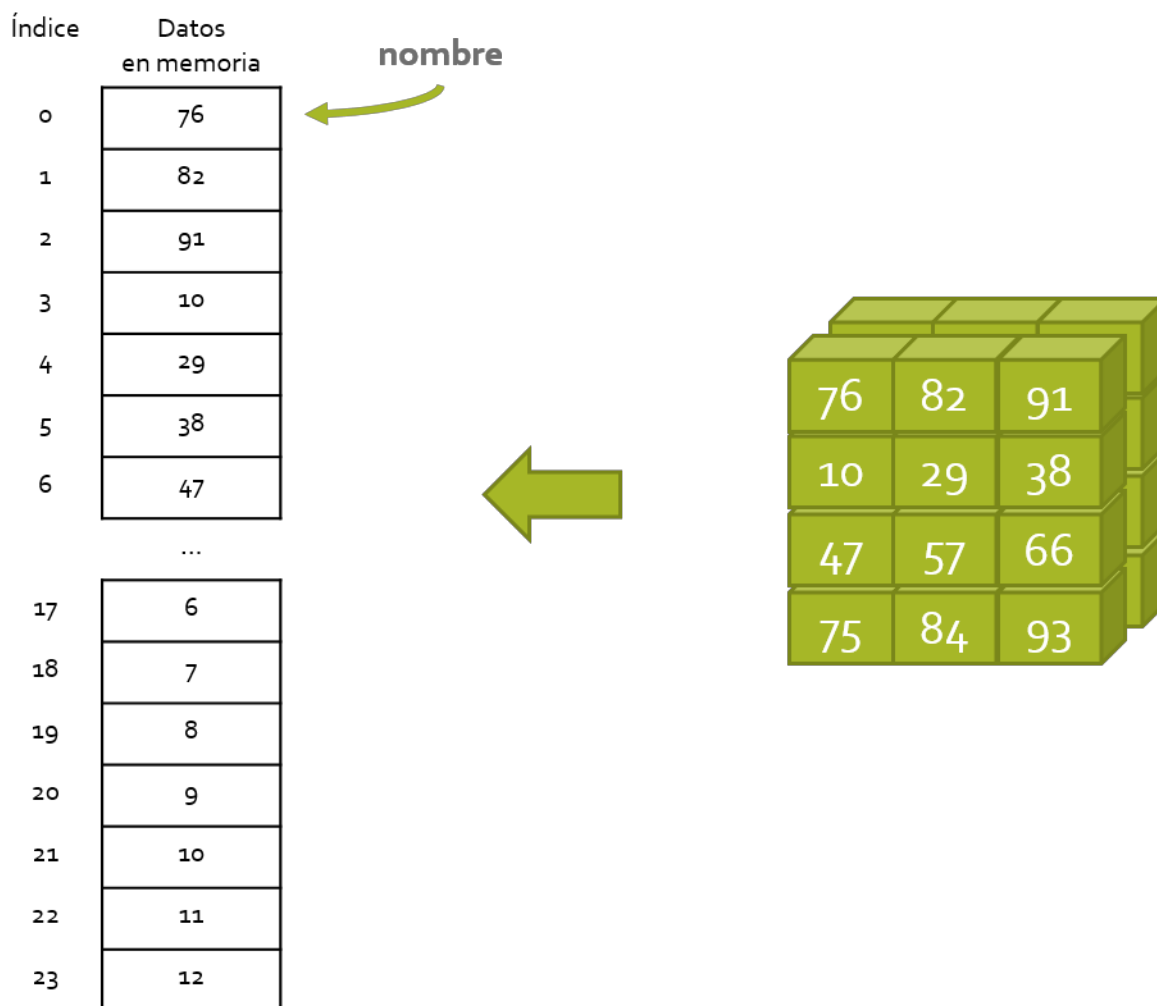



Figura 3. Representación en memoria de un arreglo tridimensional.

	Manual de prácticas del Laboratorio de Estructuras de datos y algoritmos I	Código:	MADO-19
		Versión:	01
		Página	8/151
		Sección ISO	8.3
		Fecha de emisión	20 de enero de 2017
Facultad de Ingeniería		Área/Departamento: Laboratorio de computación salas A y B	
La impresión de este documento es una copia no controlada			

Aplicaciones de arreglos

Los arreglos son una herramienta indispensable a la hora de realizar aplicaciones computacionales. Si se quiere programar un juego de mesa (como ajedrez o scrabble), llevar el control de calificaciones de un grupo de alumnos, implementar estructuras de datos, optimizar operaciones matemáticas, etc., se utilizan necesariamente arreglos.

La escítala espartana

Uno de los primeros métodos criptográficos conocidos proviene de Esparta, Grecia. El método consiste en enrollar una tira de escritura a lo largo de un palo llamado escítala y escribir sobre la tira una vez enrollada. Al desenrollar el mensaje resulta ininteligible a menos que se posea una escítala similar a la que se usó para crear el mensaje.

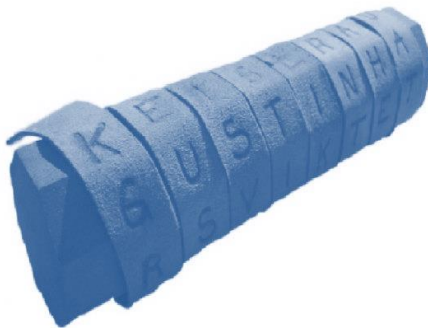



Figura 4. Forma de la escítala espartana y la tira de escritura.

	Manual de prácticas del Laboratorio de Estructuras de datos y algoritmos I	Código:	MADO-19
		Versión:	01
		Página	9/151
		Sección ISO	8.3
		Fecha de emisión	20 de enero de 2017
Facultad de Ingeniería		Área/Departamento: Laboratorio de computación salas A y B	
La impresión de este documento es una copia no controlada			

Código (la escítala espartana)

```
#include<stdio.h>


/*
Programa que realiza la implementación de la escítala espartana
Para cifrar y descifrar.
*/

void crearMensaje();
void descifrarMensaje();

int main(){
    short opcion=0;

    while (1){
        printf("\n\t*** ESCÍTALA ESPARTANA ***\n");
        printf("¿Qué desea realizar?\n");
        printf("1) Crear mensaje cifrado.\n");
        printf("2) Descifrar mensaje.\n");
        printf("3) Salir.\n");
        scanf("%d", &opcion);
        switch(opcion){
            case 1:
                crearMensaje();
                break;
            case 2:
                descifrarMensaje();
                break;
            case 3:
                return 0;
            default:
                printf("Opción no válida.\n");
        }
    }
    return 0;
}

void crearMensaje(){
    int ren, col, i, j, k=0;
    printf("Ingresar el tamaño de la escítala:\n");
    printf("\nRenglones:");
    scanf("%i",&ren);
    printf("\nColumnas:");
    scanf("%i",&col);
}
```

	Manual de prácticas del Laboratorio de Estructuras de datos y algoritmos I	Código:	MADO-19
		Versión:	01
		Página	10/151
		Sección ISO	8.3
		Fecha de emisión	20 de enero de 2017
Facultad de Ingeniería		Área/Departamento: Laboratorio de computación salas A y B	
La impresión de este documento es una copia no controlada			

```

char escitala[ren][col];
char texto[ren*col];

printf("Escriba el texto a cifrar:\n");
scanf("%s", texto);

for (i=0 ; i<ren ; i++)
    for (j=0 ; j<col ; j++)
        escitala[i][j] = texto[k++];

printf("El texto en la tira queda de la siguiente manera:\n");
for (i=0 ; i<col ; i++)
    for (j=0 ; j<ren ; j++)
        printf("%c", escitala[j][i]);

printf("\n");
}

void descifrarMensaje(){
    int ren, col, i, j, k=0;
    printf("Ingresar el tamaño de la escítala:\n");
    printf("\nRenglones:");
    scanf("%i",&ren);
    printf("\nColumnas:");
    scanf("%i",&col);


    char escitala[ren][col];
    char texto[ren*col];

    printf("Escriba el texto a descifrar:\n");
    scanf("%s", texto);

    for (i=0 ; i<col ; i++)
        for (j=0 ; j<ren ; j++)
            escitala[j][i] = texto[k++];

    printf("El texto descifrado es:\n");
    for (i=0 ; i<ren ; i++)
        for (j=0 ; j<col ; j++)
            printf("%c", escitala[i][j]);
}

```


	Manual de prácticas del Laboratorio de Estructuras de datos y algoritmos I	Código:	MADO-19
		Versión:	01
		Página	11/151
		Sección ISO	8.3
		Fecha de emisión	20 de enero de 2017
Facultad de Ingeniería		Área/Departamento: Laboratorio de computación salas A y B	
La impresión de este documento es una copia no controlada			

Bibliografía



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