	Carátula para entrega de prácticas	
Facultad de Ingeniería	Laboratorio de docencia	

Laboratorios de computación salas A y B

Profesor:

M.I. Marco Antonio Martínez Quintana

Asignatura:

Estructura de Datos y Algoritmos I

Grupo:

15

No de Práctica(s):

1

Integrante(s):

Diego Salazar Barrera

No. de Equipo de
cómputo empleado:

Na

No. de Lista o Brigada:

Na

Semestre:

2021-2

Fecha de entrega:

15/03/2021

Observaciones:

CALIFICACIÓN: _____

Objetivo:

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

Introducción:

Los arreglos son conjuntos de datos de un mismo tipo especificados por el programador desde el principio, sirven para almacenarlos en la memoria y poder acceder a ellos en cualquier momento por medio de los índices previamente declarados y siempre de forma entera. Existen arreglos unidimensionales y multidimensionales, y la única diferencia es el número de subíndices para acceder a la información. Los arreglos bidimensionales tienen forma de matriz.

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);

    char escítala[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++)
            escítala[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", escítala[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 escítala[ren][col];
    char texto[ren*col];

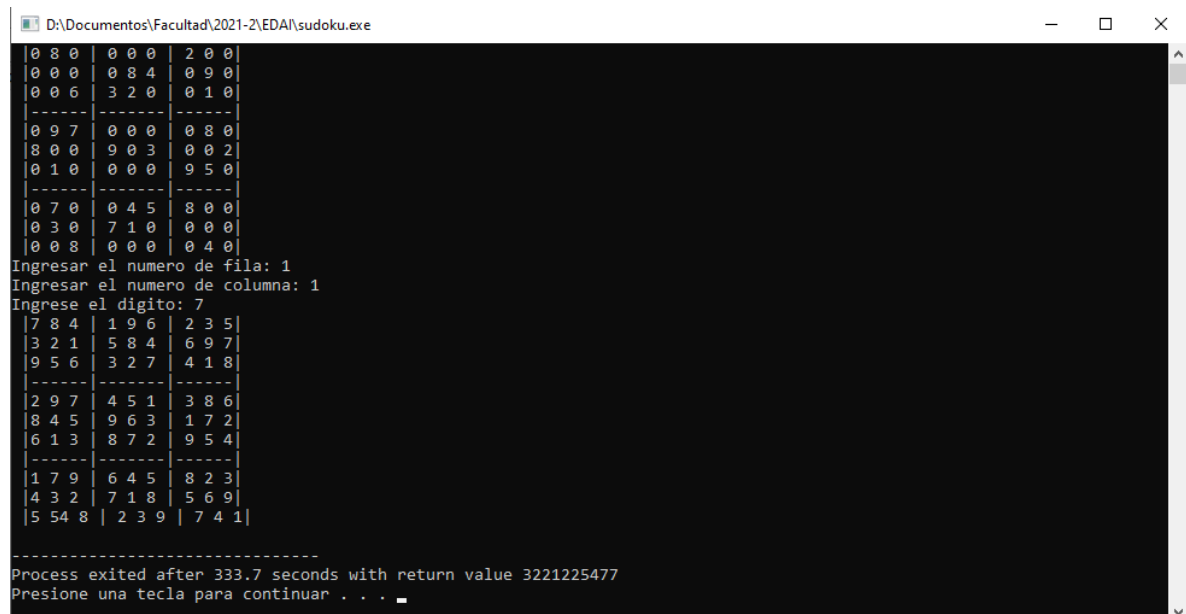
    printf("Escriba el texto a descifrar:\n");
    scanf("%s", texto);

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

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

Sudoku:

```
1 //programa elaborado por Diego Salazar Barrera
2 #include<stdio.h>
3 #include<stdlib.h>
4 #define p printf
5 #define s scanf
6 int main()
7 {
8     int a[9][9],f,c,cont=0,numero;
9     while (cont<53){
10         system("cls");
11         p(" 0 8 0 | 0 0 0 | 2 0 0|\n");
12         p(" 0 0 0 | 0 8 4 | 0 9 0|\n");
13         p(" 0 0 6 | 3 2 0 | 0 1 0|\n");
14         p("-----|-----|-----|\n");
15         p(" 0 9 7 | 0 0 0 | 0 8 0|\n");
16         p(" 8 0 0 | 9 0 3 | 0 0 2|\n");
17         p(" 0 1 0 | 0 0 0 | 9 5 0|\n");
18         p("-----|-----|-----|\n");
19         p(" 0 7 0 | 0 4 5 | 8 0 0|\n");
20         p(" 0 3 0 | 7 1 0 | 0 0 0|\n");
21         p(" 0 0 8 | 0 0 0 | 0 4 0|\n");
22         p("Ingresar el numero de fila: ");
23         s("%d",&f);
24         p("Ingresar el numero de columna: ");
25         s("%d",&c);
26         p("Ingrese el digito: ");
27         s("%d",&numero);
28         a[f][c]=numero;
29         cont++;
30         p(" |%d %d %d | %d %d %d | 2 %d %d|\n", a[1][1],a[1][3],a[1][4],a[1][5],a[1][6],a[1][8],a[1][9]);
31         p(" |%d %d %d | %d 8 4 | %d 9 %d|\n", a[2][1],a[2][2],a[2][3],a[2][4],a[2][7],a[2][9]);
32         p(" |%d %d 6 | 3 2 %d | %d 1 %d|\n", a[3][1],a[3][2],a[3][6],a[3][7],a[3][9]);
33         p(" |-----|-----|-----|\n");
34         p(" |%d 9 7 | %d %d %d | %d 8 %d|\n", a[4][1],a[4][4],a[4][5],a[4][6],a[4][7],a[4][9]);
35         p(" |8 %d %d | 9 %d 3 | %d %d 2|\n", a[5][2],a[5][3],a[5][5],a[5][7],a[5][8]);
36         p(" |%d 1 %d | %d %d %d | 9 5 %d|\n", a[6][1],a[6][3],a[6][4],a[6][5],a[6][6],a[6][9]);
37         p(" |-----|-----|-----|\n");
38         p(" |%d 7 %d | %d 4 5 | 8 %d %d|\n", a[7][1],a[7][3],a[7][4],a[7][8],a[7][9]);
39         p(" |%d 3 %d | 7 1 %d | %d %d %d|\n", a[8][1],a[8][3],a[8][6],a[8][7],a[8][8],a[8][9]);
40         p(" |%d %d 8 | %d %d %d | %d 4 %d|\n", a[9][1],a[9][2],a[9][4],a[9][5],a[9][6],a[9][7],a[9][9]);
41         return(0);
42     }
```



```
D:\Documentos\Facultad\2021-2\EDAI\sudoku.exe
0 8 0 | 0 0 0 | 2 0 0
0 0 0 | 0 8 4 | 0 9 0
0 0 6 | 3 2 0 | 0 1 0
-----|-----|-----
0 9 7 | 0 0 0 | 0 8 0
8 0 0 | 9 0 3 | 0 0 2
0 1 0 | 0 0 0 | 9 5 0
-----|-----|-----
0 7 0 | 0 4 5 | 8 0 0
0 3 0 | 7 1 0 | 0 0 0
0 0 8 | 0 0 0 | 0 4 0
Ingresar el numero de fila: 1
Ingresar el numero de columna: 1
Ingrese el digito: 7
7 8 4 | 1 9 6 | 2 3 5
3 2 1 | 5 8 4 | 6 9 7
9 5 6 | 3 2 7 | 4 1 8
-----|-----|-----
2 9 7 | 4 5 1 | 3 8 6
8 4 5 | 9 6 3 | 1 7 2
6 1 3 | 8 7 2 | 9 5 4
-----|-----|-----
1 7 9 | 6 4 5 | 8 2 3
4 3 2 | 7 1 8 | 5 6 9
5 5 4 8 | 2 3 9 | 7 4 1
-----|-----|-----
Process exited after 333.7 seconds with return value 3221225477
Presione una tecla para continuar . . .
```

Profe, una pequeña duda, no se porque en algunas ocasiones se repetía el ciclo while aunque ya hubieran pasado mas de 53 veces, me pasaba a lo mejor 1 de cada 4 veces pero me quedó la duda de que pasaba.

Conclusiones:

En lo personal creo que los arreglos son de bastante utilidad a la hora de programar, y la verdad dominarlos al 100% puede ser una gran ventaja al momento de resolver problemas, puesto que es una muy buena forma de estructurar datos y poder acceder a ellos de una forma sencilla.

En la vida cotidiana pueden ser utilizados al momento que un profesor pasa lista, puesto que cada alumno tiene una fila en específico y lo único que debe verificar es la columna correspondiente al día de clase, en cuanto a la vida escolar los arreglos son de gran ayuda al momento de realizar operaciones de matrices, puesto que tienen exactamente la misma forma y estructura y por ultimo podemos hablar del algoritmo que tienen creado los cines para escoger asiento, puesto que son filas y columnas y lo único que hace es verificar que el asiento este disponible o lleno para poder mostrarlo en pantalla.