



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):

Práctica 1: Aplicaciones de arreglos.

Integrante(s):

Escobar Flores Daniel

*No. de Equipo de cómputo
empleado:*

Semestre:

2021-2

Fecha de entrega:

15/03/2021

Observaciones:

CALIFICACIÓN: _____

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

Los arreglos nos ayudan a lograr almacenar grandes cantidades de datos de un mismo tipo, los cuales podríamos decir que van consecuentes un dato después del otro, esto con ayuda de ciclos de repetición (for) para ir recorriéndolo e ir llenándolo.

Los datos tendrán el mismo nombre de la variable con la que estén definidos y permitirán identificarse por un subíndice simplemente, el caso de ser declarados con A, se verán como A1, A2, A3, etc; así sucesivamente hasta llegar al límite que le pongamos a nuestro arreglo.

Sirven para poder manipular los datos que estén dentro, e ir comparándolos y modificándolos en dado caso, su uso más común es para hacer matrices, o algún tipo de lista de ciertos datos.

Código de escitala esparciana

```

1 //Carrera de Buss Alcan Uriel
2 //Programa que realiza la implementación de la espitola espartana Para cifrar y descifrar.
3 void crearMensaje(); //Función que en caso de crear mensaje se invocara
4 void descifrarMensaje(); //Función que en caso de descifrar mensaje se invocara
5 int main() { //Inicio de nuestro método principal
6     short opcion; //Variable que validare el switch
7
8     while(1) { // Ciclo de repetición hasta que el usuario decida no ejecutar mas veces el programa
9         printf("\n\t\t\t\t\t ESPITALA ESPARTANA ****\n");
10        printf("\n\t\t\t\t\t Que desea realizar\n");
11        printf("\t\t\t\t\t 1) Crear mensaje cifrado.\n"); // Menú mostrado en pantalla//
12        printf("\t\t\t\t\t 2) Descifrar mensaje.\n");
13        printf("\t\t\t\t\t 3) Salir.\n");
14        scanf("%d",&opcion); // Asignación a la variable opcion por parte del usuario
15        switch(opcion){ // Se valen el alor de la variable y entra al caso correspondiente
16            case 1: crearMensaje(); //Invoca a la función crearMensaje
17                break; //Sale de la estructura de control switch
18            case 2: descifrarMensaje(); //Invoca a la función descifrarMensaje
19                break; //Sale de la estructura de control switch
20            case 3: return 0; //Finaliza el programa
21            default: printf("Opcion no válida.\n"); // En caso de no existir el caso mandara este mensaje
22        }
23        //Fin del switch
24        //Fin del ciclo while
25    }
26    //Fin del metodo principal
27
28    void crearMensaje(){ //Función crearMensaje
29        int row,col,i,j,k; //Variables de tipo entero
30        printf("\t\t\t\t\t Ingrese el tamaño de la espitala:\n");
31        printf("\t\t\t\t\t Ingrese:\n");

```

```

30 scanf("%i",&ren); //Numero de rengalones ingresadas por el usuario almacenar el mensaje
31 printf("\nColumnas:");
32 scanf("%i",&col); //Numero de columnas ingresadas por el usuario almacenar el mensaje
33 char escitala[ren][col]; //Arreglo de tipo caracter
34 char texto[ren*col]; //Arreglo unidimensional, bidimensional
35 printf("Escriba el texto a cifrar:\n");
36 scanf("%s",texto); //Texto ingresado por el usuario para almacenarlo en la variable texto
37
38 for(i=0;i<ren;i++) //Ciclo de repetición for
39 for(j=0;j<col;j++) //Ciclo de repetición for
40 escitala[i][j]=texto[k++];
41 printf("El texto en la tira queda de la siguiente manera:\n");
42 for(i=0;i<col;i++) //Ciclo de repetición for
43 for(j=0;j<ren;j++) //Ciclo de repetición for
44 printf("%c",escitala[j][i]);
45 printf("\n");
46 } //Fin de función
47
48 void descifrarMensaje() { //Función descifrarMensaje
49 int ren,col,i,j,k=0; //Variables de tipo entero
50 printf("Ingresar el tamaño de la escitala:\n");
51 printf("\nReinglones:");
52 scanf("%i",&ren); //Numero de rengalones ingresadas por el usuario almacenar el mensaje
53 printf("\nColumnas:");
54 scanf("%i",&col); //Numero de columnas ingresadas por el usuario almacenar el mensaje
55
56 char escitala[ren][col];
57 char texto[ren*col];

```

```

Reinglones:4
Columnas:5
Escriba el texto a cifrar:
licomovaelplandehoy?
El texto en la tira queda de la siguiente manera:
lopeivlhcaaoenymld?

*** ESCITALA ESPARTANA ***
Que desea realizar?
) Crear mensaje cifrado.
) Descifrar mensaje.
) Salir.

Ingresar el tamaño de la escitala:
Reinglones:4
Columnas:5
Escriba el texto a descifrar:
lopeivlhcaaoenymld?
El texto descifrado es:
licomovaelplandehoy?

*** ESCITALA ESPARTANA ***
Que desea realizar?
) Crear mensaje cifrado.
) Descifrar mensaje.
) Salir.

.....
Process exited with return value 0
Press any key to continue . . .

```

Sudoku código.

```

1 #include <iostream>
2 #include <conio.h>
3 #include <stdio.h>
4 #include <stdlib.h>
5 //Carrasco Ruiz Alan Uriel
6 int M[9][9], filas=9, columnas=9;
7 void imprimir();
8
9 main() {
10 printf("Bienvenido al juego Sudoku\n");
11 printf("Ingresa valores del 1 al 9 solamente\n");
12 printf("Si el valor ya esta asignado vuelve a introducirlo\n");
13 system("pause");
14 M[0][2] = 0; //Asignación de algunos valores de la matriz
15 M[0][3] = 5;
16 M[0][4] = 4;
17 M[0][5] = 7;
18 M[1][3] = 6;
19 M[1][2] = 5;
20 M[1][4] = 7;
21 M[2][0] = 4;
22 M[2][5] = 3;
23 M[2][6] = 6;
24 M[3][1] = 6;
25 M[3][5] = 8;
26 M[3][6] = 9;
27 M[4][2] = 7;
28 M[4][3] = 8;
29 M[4][5] = 5;

```

```

31 M[0][2] = 1;
32 M[0][4] = 0;
33 M[0][7] = 7;
34 M[7][1] = 0;
35 M[7][2] = 0;
36 M[7][8] = 0;
37 M[7][4] = 0;
38 M[0][8] = 7;
39
40 for(int i=0; i<filas; i++){
41     for (int j=0; j<columnas; j++){
42         imprimir();
43         printf("Dame un numero para el lugar [%d,%d]",i,j);
44         scanf("%d",&M[i][j]);
45     }
46     imprimir();
47 }
48
49 printf("Felicitades si ningún numero se repitio has ganado\n");
50 getch();
51 return 0;
52 }
53
54 void imprimir(){
55     system("cls");
56     for (int i = 0; i < 9; i++){
57         for (int j = 0; j < 9; j++){
58             printf("%d|",M[i][j]);
59         }
60     }
61 }

```

C:\Users\aur2\Dev C\Dev-Cpp\ConsolePauser.exe

0	0	3	5	0	0	4	0	7
0	6	5	0	7	0	1	0	0
4	0	0	0	0	3	6	0	0
0	4	0	0	0	8	3	0	0
0	0	7	9	0	5	0	0	0
0	0	0	0	0	0	0	0	0
0	8	1	0	6	0	0	7	0
0	5	9	2	4	0	0	0	0
0	0	0	0	0	7	0	0	0

Dame un numero para el lugar [0,0]

1	2	3	5	8	6	4	9	7
9	6	5	4	7	2	1	3	8
4	7	8	1	9	3	6	5	2
0	4	0	0	0	8	3	0	0
0	0	7	9	0	5	0	0	0
0	0	0	0	0	0	0	0	0
0	8	1	0	6	0	0	7	0
0	5	9	2	4	0	0	0	0
0	0	0	0	0	7	0	0	0

Dame un numero para el lugar [3,0]

Conclusión:

Esta practica es muy importante, ya que nos ayuda a la implementación de datos, en lo general me costo trabajo poder obtener los resultados esperados ya que no tenía claro cómo era la operación de sudoku. forma de matriz ya que nos delimitaba los tamaños.