



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: Marco Antonio Martínez Quintana

Asignatura: Estructura de Datos y Algoritmos I

Grupo: 15

No de Práctica(s): 0

Integrante(s): Perera Martínez David

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

No. de Lista o Brigada:

Semestre: 2021-2022

Fecha de entrega: 15 de marzo de 2021

Observaciones:

CALIFICACIÓN: _____

Objetivo:

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

Introducción:

Las disposiciones nos permiten ahorrar diferentes variables del mismo tipo en una unidad de memoria continuamente, gracias a esto, se permite acceder de manera rápida y efectiva.

Como se mencionó anteriormente, todos los datos almacenados en un acuerdo son de un solo tipo, si son completos, caracteres, flotadores, etc.

Actualmente, hemos utilizado acuerdos individuales y bidimensionales, unidimensional, ya que su nombre nos dice que son una dimensión, es decir, tendremos una sola fila con diferentes gases. Si bien el bidimensional ya ha podido tener más de filas, están definidos por el usuario o ya están definidos.

Desarrollo:

Programa del manual de prácticas

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);
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){ //seleccionador para elegir que opcion desea realizar
case 1:
crearMensaje(); //manda a llamar a la función crear mensaje
break;
case 2:
descifrarMensaje(); // manda a llamar a la función decifrar mensaje
break;
case 3:
return 0;
default:
printf("Opción no válida.\n");
}
}
return 0;
}

void crearMensaje(){ // cuerpo de la función para crear el mensaje
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 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++)

```

Ejecución del programa

```

*** ESC=TALA ESPARTANA ***
¿Qu  desea realizar?
1) Crear mensaje cifrado.
2) Descifrar mensaje.
3) Salir.
1
Ingresar el tama o de la esc tala:

Ren ones:4

Columnas:3
Escriba el texto a cifrar:
ssammysalio
El texto en la tira queda de la siguiente manera:
smsismaoayl

```

```
*** ESC=TALA ESPARTANA ***
¿Què desea realizar?
1) Crear mensaje cifrado.
2) Descifrar mensaje.
3) Salir.
```

No hubo necesidad de corregir nada ya que me corrió a la perfección todo el código que estaba.

Actividad (Sudoku):

```
#include <stdio.h>

int main() {
    printf("\n\t\t\tSUDOKU\n\n");
    int Sudoku[9][9] = {3,0,0,0,0,0,0,0,0,
                        0,0,0,0,0,0,0,0,0,
                        0,0,0,0,0,0,0,0,0,
                        0,0,0,0,0,0,0,0,0,
                        0,0,0,0,0,0,0,0,0,
                        0,0,0,0,0,0,0,0,0,
                        0,0,0,0,0,0,0,0,0,
                        0,0,0,0,0,0,0,0,0,
                        0,0,0,0,0,0,0,0,0};

    int i,j;
    for(i=0;i<9;i++){
        for(j=0;j<9;j++){
            printf("[%d] ",Sudoku[i][j]);
        }
        printf("\n");
    }

    printf("Complete las espacios con 0 del sudoku\n\n");

    int x;

    printf("posicion [1,1] :"); scanf("%d",&x); Sudoku[0][1]=x;
    printf("posicion [1,2] :"); scanf("%d",&x); Sudoku[0][2]=x;
    printf("posicion [1,4] :"); scanf("%d",&x); Sudoku[0][3]=x;
    printf("posicion [1,5] :"); scanf("%d",&x); Sudoku[0][4]=x;
    printf("posicion [1,6] :"); scanf("%d",&x); Sudoku[0][5]=x;
    printf("posicion [1,9] :"); scanf("%d",&x); Sudoku[0][8]=x;
    printf("\n");

    printf("posicion [2,6] :"); scanf("%d",&x); Sudoku[1][5]=x;
    printf("posicion [2,7] :"); scanf("%d",&x); Sudoku[1][6]=x;
    printf("posicion [2,8] :"); scanf("%d",&x); Sudoku[1][7]=x;
    printf("posicion [2,9] :"); scanf("%d",&x); Sudoku[1][8]=x;
    printf("\n");

    printf("posicion [3,1] :"); scanf("%d",&x); Sudoku[2][0]=x;
```

[illegible]

```

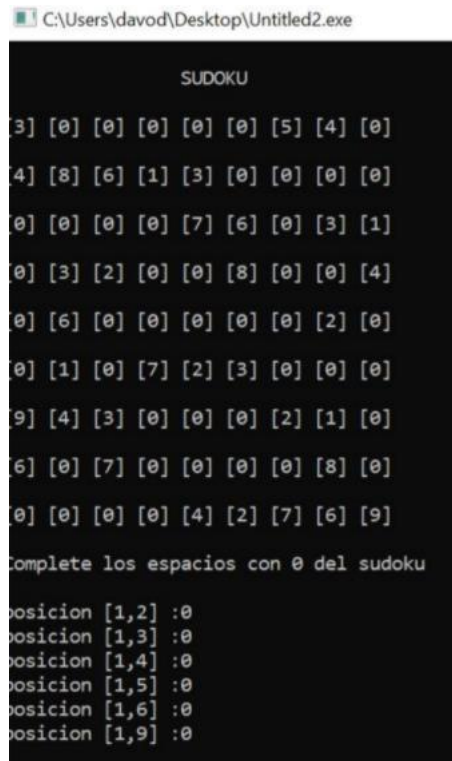
printf("posicion [8,4] :"); scanf("%d",&x); Sudoku[7][3]=x;
printf("posicion [8,5] :"); scanf("%d",&x); Sudoku[7][4]=x;
printf("posicion [8,6] :"); scanf("%d",&x); Sudoku[7][5]=x;

printf("posicion [9,4] :"); scanf("%d",&x); Sudoku[8][3]=x;
printf("\n");

for(i=0;i<9;i++){
    for(j=0;j<9;j++){
        printf("[%d] ",Sudoku[i][j]);
    }
    printf("\n\n");
}
system("pause");

```

Ejecución del programa



Así es como llegamos a la ejecución del sudoku

Conclusión

Esta práctica se ha convertido en algo complicado principalmente porque no sabía cómo la operación de Sudoku, hasta que tuve que explorar, como era como y podría tener práctica con algunos fracasos, porque ya he olvidado ciertas cosas durante la programación, pero en el final llegó. Utilice y refuerce el arreglo multidimensional que se ha hecho una práctica muy buena e interesante para este tema.