

En esta práctica se va a utilizar matrices, ficheros, operadores de bits, argumentos del main y compilación por separado.

Básicamente consiste en rotar los bits de los elementos de una matriz de  $M \times N$  letras mayúsculas (siendo  $M$  y  $N$  constantes definidas previamente): los dos bits de menor peso (más a la derecha) pasarán a ser los dos de mayor peso (más a la izda). Para realizar la práctica se deberán implementar las siguientes funciones:

- ***Imprimir\_mat***, que recibiendo como parámetro una matriz como la mencionada, la imprima en pantalla.
- ***Rellenar\_mat***, que recibiendo como parámetro una matriz de  $M \times N$  letras mayúsculas, la rellene con valores generados aleatoriamente.
- ***Rotar\_mat***, que pasándole como parámetro una matriz como la mencionada, rote los dos bits de más a la dcha de cada entero de la matriz, de forma que estos dos pasen a ser los de más a la izda después de desplazar el resto dos posiciones hacia la dcha.
- ***Desrotar\_mat***, que pasándole como parámetro una matriz como la mencionada, haga lo contrario que la función anterior, es decir, devuelva los dos bits movidos a su posición original.

Todas estas funciones se implementarán en un fichero .c de compilación por separado. Además se implementará un fichero .h en el que se definirán las constantes  $M$  y  $N$  y se incluirán los prototipos de las funciones descritas.

Por último, se realizará un programa principal que recibirá como parámetro el nombre de un fichero y realizará las siguientes acciones:

- a) Abrir el fichero.
- b) Rellenar la matriz.
- c) Imprimir la matriz.
- d) Grabar la matriz en el fichero.
- e) Rotar los bits de cada elemento de la matriz como se ha explicado anteriormente.
- f) Imprimir la matriz modificada.
- g) Grabar la matriz en el fichero.
- h) Desrotar los bits de cada elemento de la matriz como para devolverlos a su estado inicial.
- i) Imprimir la matriz.
- j) Cerrar el fichero.

Una posible ejecución para una matriz de 15x10 sería la siguiente:

```

P L U N Z P S Q Z S X D O V N
H N S Q U G I P N R T P E G P
Q L C D S G M E I E L O J P L
G F X U R R X W U M N S O U O J
H D G X T R C I K Z C E U E U
U X G W K P Y K K K V F P V R
M F B F J N S E A V P X U L U
E I W N P R Y F S C S Z A F I
A R K F R U Y F U X F B B P P
V C F O E Y C M W F A C O B V

Q W ! ! o o u Q E T i o E
t o E T o D S Q o S Q Q D Q
T ! ! o A E D S Q R Q H E Q ! !
D æ ! U o ! o S o E E U E æ
+ A D ! S o o R E E U D Q o Q o
U ! D ! E Q U E E E U æ Q U o
S æ E æ æ o E Q P U Q ! o ! ! o
Q R ! o Q o U æ E o E o P æ R
P o E æ o o U æ o ! æ E E Q Q
U o æ E Q U o S ! æ P o E E U

P L U N Z P S Q Z S X D O V N
H N S Q U G I P N R T P E G P
Q L C D S G M E I E L O J P L
G F X U R R X W U M N S O U O J
H D G X T R C I K Z C E U E U
U X G W K P Y K K K V F P V R
M F B F J N S E A V P X U L U
E I W N P R Y F S C S Z A F I
A R K F R U Y F U X F B B P P
V C F O E Y C M W F A C O B V

```

**Solución:**

Fichero .h

```
#ifndef ROTAR_H
#define ROTAR_H

#define N 10
#define M 15

void imprimir(char[M][N]);
void rellenar(char[M][N]);
void rotar(char[M][N]);
void desrotar(char[M][N]);
void matToFile(char[M][N], FILE*);
int randomNum(int, int);

#endif
```

Fichero .c (de funciones)

```
#include <time.h>
#include <stdlib.h>
#include <stdio.h>
#include "rotar.h"

void rellenar(char mat[M][N]){
    int i= 0;
    int j = 0;

    for (i = 0; i < M; i++){
        for (j = 0; j < N; j++){
            mat[i][j] = randomNum('A', 'Z');
        }
    }
}

void imprimir(char mat[M][N]){
    int i = 0;
    int j = 0;

    printf("\n");
    for (i = 0; i < M; i++){
        for (j = 0; j < N; j++){
```

```
        printf(" %c ", mat[i][j]);
    }
    printf("\n");
}

void rotar(char mat[M][N]){
    int i = 0;
    int j = 0;

    for (i = 0; i < M; i++){
        for(j= 0; j < N; j++){
            mat[i][j] = ( (mat[i][j] & 3) << 6 ) | (mat[i][j] >> 2);
        }
    }
}

void desrotar (char mat[M][N])
{
    int i = 0;
    int j = 0;

    for (i=0; i<M; i++){
        for(j=0; j<N; j++){
            mat[i][j] = ((mat[i][j] & 192) >> 6 ) | (mat[i][j] << 2);
        }
    }
}

void matToFile(char mat[M][N], FILE* fichOut){
    int i = 0;

    for (i = 0; i < M; i++)
        fputs(mat[i], fichOut);
}

int randomNum(int min, int max){
    return (rand() % (max - min + 1) + min);
}
```

Fichero .c (del main)

```
#include <stdio.h>
#include <stdlib.h>
```

```
#include "rotar.h"

int main(int argc, char* argv[])
{
    FILE* fichOut;
    char mat[M][N];

    if (argc != 2){
        printf("ERROR: Maximo dos argumentos");
        return 1;
    }

    if (!(fichOut = fopen(argv[1], "w"))){
        printf("ERROR: No ha sido posible abrir/crear el fichero");
        return 2;
    }

    rellenar(mat);
    imprimir(mat);
    matToFile(mat, fichOut);
    rotar(mat);
    imprimir(mat);
    matToFile(mat, fichOut);
    desrotar(mat);
    imprimir(mat);
    matToFile(mat, fichOut);

    fclose(fichOut);

    return 0;
}
```