#ifndef \_COMMON\_H\_

#define \_COMMON\_H\_

#include <stdbool.h>

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* En este archivo se definen todas las estructuras, variables y códigos de salida

\* que se usan en toda la aplicación.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* --------------------------- \*/

/\* DEFINICIONES DE ESTRUCTURAS \*/

/\* --------------------------- \*/

/\*\*

\* Estructura para las dimensiones de la matriz

\*/

typedef struct {

int filas;

int columnas;

} Dim;

/\*\*

\* Estructura Diamante

\*/

typedef struct {

//id = {1,2,3,4,5,6,7,8} indican el tipo de diamante

//id = {0} indica que no hay fiamante en ese hueco

int id;

const char \*img;

} Diamante;

/\*\*

\* Estructura para la matriz de juego

\*/

typedef struct {

int nivel; /\* Nivel actual \*/

Dim dimens; /\* Dimensiones de la matriz \*/

Diamante \*matriz; /\* Matriz de juego \*/

} Malla;

/\* ------------------------- \*/

/\* DEFINICIONES DE ETIQUETAS \*/

/\* ------------------------- \*/

/\*\*

\* Nivel máximo en el juego

\*/

#define MAX\_NV 3

/\*\*

\* Valor para representar un hueco en el tablero (diamante vacío)

\*/

#define DIAMANTE\_VACIO 0

/\*\*

\* Mayor valor admitido para un diamante

\*/

#define DIAMANTE\_MAX 8

/\* ------------------------------ \*/

/\* CÓDIGOS DE SALIDA DE FUNCIONES \*/

/\* ------------------------------ \*/

/\*\*

\* Con cambios a la hora de recorrer la columna para tratar los huecos.

\*/

#define CON\_CAMBIOS 2

/\*\*

\* Sin cambios a la hora de recorrer la columna para tratar los huecos.

\*/

#define SIN\_CAMBIOS 1

/\*\*

\* Éxito al procesar un argumento que requiere que se termine la ejecución (por ejemplo,

\* la opción '-h'.

\*/

#define SUCC\_ARGS 1

/\*\*

\* La función acabó sin problemas.

\*/

#define SUCCESS 0

/\*\*

\* Error al pasar los argumentos a la función.

\*/

#define ERR\_ARGS -1

/\*\*

\* Error al abrir un archivo.

\*/

#define ERR\_ARCHIVO -2

/\*\*

\* Error al gestionar memoria (reservar, liberar...).

\*/

#define ERR\_MEM -3

/\*\*

\* Error producido en alguna función de CUDA.

\*/

#define ERR\_CUDA -4

/\*\*

\* Error al intentar utilizar una matriz que no cabe en el dispositivo.

\*/

#define ERR\_TAM -5

#endif