

```

//
// Created by Antonio Rodríguez Rodríguez y José Antonio Carmona Molina on 6/10/21.
// 2º Ingeniería Informática + ADE
//

#ifndef PRACTICA2_SUDOKU_H
#define PRACTICA2_SUDOKU_H

#endif

#include <iostream>

using namespace std;

class Sudoku {
private:
    int nfilas = 9;
    int ncols = 9;
    int sudoku[nfilas][ncols]; // El sudoku es una matriz de enteros.
    int valor; // Valor de cada casilla.

public:
    /**
     * @brief Constructor por defecto.
     */
    Sudoku();

    /**
     * @brief Constructor con valores iniciales.
     * @param valores
     * @pre Valor máximo <= 24. // Ya que como máximo se asigna el valor a 3 casillas que al tener
que tener valores diferentes,
     * pueden ser 9,8,7; cuya suma nunca será mayor que 24.
     * @post El sudoku generado tiene los valores iniciales que se asignan en función al
comportamiento del constructor.
     */
    Sudoku(int * valores);

    /**
     * @brief Constructor de copia.
     * @param orig
     */
    Sudoku(const Sudoku & orig);

    /**
     * @brief Devuelve el valor de una posición determinada.
     * @param i fila de la que se quiere obtener el valor.
     * @param j columna de la que se quiere obtener el valor.
     * @return Elemento de la posición indicada.
     * @pre param >= 0 && param <= 9.
     * @pre El valor no se modifica.
     */

```

```

int get_value(int i, int j) const;

/**
 * @brief Devuelve el valor de una posición determinada.
 * @param i fila de la que se quiere obtener el valor.
 * @param j columna de la que se quiere obtener el valor.
 * @return Elemento de la posición indicada.
 * @pre param >= 0 && param <= 9.
 * @post El valor puede haberse modificado.
 */
int get_value(int i, int j);

/**
 * @brief Inserta un valor en una posición determinada.
 * @param i fila de la que se quiere insertar el valor.
 * @param j columna de la que se quiere insertar el valor.
 * @return Si se ha insertado el valor correctamente.
 * @pre param >= 0 && param <= 9.
 */
bool set_value(int i, int j);

/**
 * @brief Este método comprueba que no se repitan valores dentro de cada cuadrado pequeño.
 * @return Si se cumple.
 */
bool Comprobar_recuadro() const;

/**
 * @brief Comprueba si hay valores repetidos en las filas del sudoku.
 * @return Si se cumple.
 */
bool Repetidos_fila() const;

/**
 * @brief Comprueba si hay valores repetidos en las columnas del sudoku.
 * @return Si se cumple.
 */
bool Repetidos_cols() const;

/**
 * @brief Resuelve el sudoku.
 * @return Devuelve el sudoku resuelto.
 * @post El sudoku se devuelve modificado.
 */
Sudoku & Resolver_Sudoku(); // Utilizaría las funciones anteriores en este método.

};

```