

## TDA Sudoku Killer

Generado por Doxygen 1.8.17



<b>1 TDA Sudoku Killer</b>	<b>1</b>
<b>2 Índice de clases</b>	<b>3</b>
2.1 Lista de clases . . . . .	3
<b>3 Documentación de las clases</b>	<b>5</b>
3.1 Referencia de la Clase jaula . . . . .	5
3.1.1 Descripción detallada . . . . .	5
3.1.2 Documentación del constructor y destructor . . . . .	5
3.1.2.1 jaula() . . . . .	6
3.1.3 Documentación de las funciones miembro . . . . .	6
3.1.3.1 get_horizontal() . . . . .	6
3.1.3.2 get_inicio() . . . . .	6
3.1.3.3 get_longitud() . . . . .	7
3.1.3.4 get_valor() . . . . .	7
3.2 Referencia de la Clase sudoku_killer . . . . .	7
3.2.1 Descripción detallada . . . . .	8
3.2.2 Documentación del constructor y destructor . . . . .	8
3.2.2.1 sudoku_killer() [1/2] . . . . .	8
3.2.2.2 sudoku_killer() [2/2] . . . . .	8
3.2.3 Documentación de las funciones miembro . . . . .	8
3.2.3.1 columna_correcta() . . . . .	8
3.2.3.2 cuadrícula_correcta() . . . . .	9
3.2.3.3 get_valor() . . . . .	9
3.2.3.4 jaula_correcta() . . . . .	10
3.2.3.5 sudoku_correcto() . . . . .	10
<b>Índice alfabético</b>	<b>11</b>



# Capítulo 1

## TDA Sudoku Killer

Autor

- Sergio Salvador Gil



## Capítulo 2

# Índice de clases

### 2.1. Lista de clases

Lista de las clases, estructuras, uniones e interfaces con una breve descripción:

<a href="#">jaula</a>	Clase que representa una jaula del sudoku killer . . . . .	5
<a href="#">sudoku_killer</a>	Clase que representa un sudoku killer . . . . .	7





## Capítulo 3

# Documentación de las clases

### 3.1. Referencia de la Clase jaula

clase que representa una jaula del sudoku killer.

```
#include <SudokuKiller.h>
```

#### Métodos públicos

- `jaula` (coord \_inicio, int \_valor, int \_longitud, bool \_horizontal)  
*Constructor para la clase jaula a partir de su valor, su longitud y su dirección.*
- `jaula` ()  
*constructor sin parámetros de la clase jaula, que inicializa un objeto con longitud 0 (valor tambien se inicializa a 0).*
- coord `get_inicio` () const
- int `get_longitud` () const
- int `get_valor` () const
- bool `get_horizontal` () const

#### 3.1.1. Descripción detallada

clase que representa una jaula del sudoku killer.

una jaula está representada con una posición de inicio, una dirección (horizontal, expandiéndose hacia la derecha, o vertical, si no es horizontal y se desplaza hacia abajo), y su longitud. Además, cada jaula tiene un valor, que es correcto si es el resultado de la suma de los valores de todos sus elementos.

#### Precondición

longitud  $\geq 0$

valor  $> 0$

#### 3.1.2. Documentación del constructor y destructor

### 3.1.2.1. `jaula()`

```
jaula::jaula (
    coord _inicio,
    int _valor,
    int _longitud,
    bool _horizontal )
```

Constructor para la clase `jaula` a partir de su valor, su longitud y su dirección.

#### Parámetros

<code>_valor,el</code>	valor de la jaula
<code>_longitud</code>	La longitud del array de coordenadas.

#### Precondición

`_longitud >= 0.`

#### Postcondición

se construye el objeto de la clase `jaula` con los datos proporcionados.

### 3.1.3. Documentación de las funciones miembro

#### 3.1.3.1. `get_horizontal()`

```
bool jaula::get_horizontal ( ) const
```

#### Devuelve

horizontal

#### 3.1.3.2. `get_inicio()`

```
coord jaula::get_inicio ( ) const
```

#### Devuelve

inicio

### 3.1.3.3. get\_longitud()

```
int jaula::get_longitud ( ) const
```

Devuelve

longitud

### 3.1.3.4. get\_valor()

```
int jaula::get_valor ( ) const
```

Devuelve

valor

La documentación para esta clase fue generada a partir del siguiente fichero:

- SudokuKiller.h

## 3.2. Referencia de la Clase sudoku\_killer

clase que representa un sudoku killer.

```
#include <SudokuKiller.h>
```

### Métodos públicos

- `sudoku_killer ()`  
*constructor sin parámetros*
- `sudoku_killer (const int *valores, const jaula *_jaulas, int _num_jaulas)`  
*constructor del sudoku killer con un array de valores y uno de jaulas*
- `int get_valor (int fila, int columna)`  
*devuelve el valor de una casilla del sudoku killer*
- `bool cuadrricula_correcta (coord subcuadrícula)`  
*Comprueba si la subcuadrícula dada es correcta.*
- `bool columna_correcta (int columna)`  
*Comprueba si la columna dada es correcta.*
- `bool jaula_correcta (const jaula &jaula)`  
*comprueba que una jaula del sudoku killer sea correcta*
- `bool sudoku_correcto ()`  
*comprueba si el sudoku\_killer está resuelto correctamente.*

### 3.2.1. Descripción detallada

clase que representa un sudoku killer.

cada sudoku killer es una cuadrícula de  $9 \times 9$  que tiene a su vez 9 subcuadrículas  $3 \times 3$ . cada celda del sudoku killer está contenida en una jaula, cada una con un valor especificado que es el resultado de la suma de los valores de todas sus celdas.

#### Precondición

todas las casillas tienen que estar incluídas en una jaula y solo una.

$1 \leq \text{valores} \leq 9$ .

### 3.2.2. Documentación del constructor y destructor

#### 3.2.2.1. `sudoku_killer()` [1/2]

```
sudoku_killer::sudoku_killer ( )
```

constructor sin parámetros

inicializa el objeto con todas las casillas con valor 0, y sin jaulas.

#### 3.2.2.2. `sudoku_killer()` [2/2]

```
sudoku_killer::sudoku_killer (
    const int * valores,
    const jaula * _jaulas,
    int _num_jaulas )
```

constructor del sudoku killer con un array de valores y uno de jaulas

#### Parámetros

<i>valores</i>	array con los valores del <code>sudoku_killer</code>
----------------	--

### 3.2.3. Documentación de las funciones miembro

#### 3.2.3.1. `columna_correcta()`

```
bool sudoku_killer::columna_correcta (
    int columna )
```

Comprueba si la columna dada es correcta.

**Parámetros**

<i>columna</i>	representa el índice de la columna que se quiere comprobar
----------------	--

Una columna es correcta si está llena y cada uno de los número se encuentra una única vez.

**Devuelve**

true si la scolumna es correcta

**Precondición**

$0 \leq \text{columna} \leq 9$

**3.2.3.2. cuadrícula\_correcta()**

```
bool sudoku_killer::cuadrícula_correcta (
    coord subcuadrícula )
```

Comprueba si la subcuadrícula dada es correcta.

**Parámetros**

<i>subcuadrícula</i>	representa una de las subcuadrículas de la cuadrícula del sudoku. Cada subcuadrícula se representa con una coordenada como si fuera una celda de un sudoku 3 * 3.
----------------------	---

Una subcuadrícula es correcta si está llena y cada uno de los número se encuentra una única vez.

**Devuelve**

true si la subcuadrícula es correcta

**Precondición**

$0 \leq \text{subcuadrícula.fila}, \text{subcuadrícula.columna} \leq 3$

**3.2.3.3. get\_valor()**

```
int sudoku_killer::get_valor (
    int fila,
    int columna )
```

devuelve el valor de una casilla del sudoku killer

**Precondición**

$0 \leq \text{fila} \leq 9$

$0 \leq \text{columna} \leq 9$

**Devuelve**

valores [fila][columna]

**3.2.3.4. jaula\_correcta()**

```
bool sudoku_killer::jaula_correcta (
    const jaula & _jaula )
```

comprueba que una jaula del sudoku killer sea correcta

una jaula es correcta si las componentes suman su valor. Por ejemplo, una jaula con valor 10, que tenga como inicio la casilla (0,0) y de longitud horizontal 2, es correcta únicamente si `valores[0][0] + valores[0][1] = 10`

**Devuelve**

true si la jaula es correcta

**3.2.3.5. sudoku\_correcto()**

```
bool sudoku_killer::sudoku_correcto ( )
```

comprueba si el [sudoku\\_killer](#) está resuelto correctamente.

El [sudoku\\_killer](#) es correcto únicamente si todas sus columnas son correctas, todas sus subcuadrículas son correctas, y además si todas sus jaulas son también correctas.

**Devuelve**

true si el sudoku es correcto.

La documentación para esta clase fue generada a partir del siguiente fichero:

- SudokuKiller.h

# Índice alfabético

- columna\_correcta
  - sudoku\_killer, [8](#)
- cuadrícula\_correcta
  - sudoku\_killer, [9](#)
- get\_horizontal
  - jaula, [6](#)
- get\_inicio
  - jaula, [6](#)
- get\_longitud
  - jaula, [6](#)
- get\_valor
  - jaula, [7](#)
  - sudoku\_killer, [9](#)
- jaula, [5](#)
  - get\_horizontal, [6](#)
  - get\_inicio, [6](#)
  - get\_longitud, [6](#)
  - get\_valor, [7](#)
  - jaula, [5](#)
- jaula\_correcta
  - sudoku\_killer, [10](#)
- sudoku\_correcto
  - sudoku\_killer, [10](#)
- sudoku\_killer, [7](#)
  - columna\_correcta, [8](#)
  - cuadrícula\_correcta, [9](#)
  - get\_valor, [9](#)
  - jaula\_correcta, [10](#)
  - sudoku\_correcto, [10](#)
  - sudoku\_killer, [8](#)