

My Project

Generated by Doxygen 1.8.11

Contents

1	Class Index	1
1.1	Class List	1
2	File Index	3
2.1	File List	3
3	Class Documentation	5
3.1	elemento Struct Reference	5
3.2	Pila_max Class Reference	5
3.2.1	Detailed Description	6
3.2.2	Constructor & Destructor Documentation	7
3.2.2.1	Pila_max(const Pila_max &pm)	7
3.2.3	Member Function Documentation	7
3.2.3.1	operator!=(const Pila_max &pm)	7
3.2.3.2	operator<(const Pila_max &pm)	7
3.2.3.3	operator=(const Pila_max &pm)	7
3.2.3.4	operator==(const Pila_max &pm)	7
3.2.3.5	operator>(const Pila_max &pm)	8
3.2.3.6	push(int e)	8
4	File Documentation	9
4.1	pila_max_vd.h File Reference	9
4.1.1	Detailed Description	9
	Index	11

Chapter 1

Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

elemento	5
Pila_max		
T.D.A. Pila_max	5

Chapter 2

File Index

2.1 File List

Here is a list of all documented files with brief descriptions:

pila_max_vd.h	Fichero cabecera del TDA Pila max	9
-------------------------------	---	---

Chapter 3

Class Documentation

3.1 elemento Struct Reference

Public Attributes

- int **ele**
- int **maximo**

The documentation for this struct was generated from the following file:

- [pila_max_vd.h](#)

3.2 Pila_max Class Reference

T.D.A. [Pila_max](#).

```
#include <pila_max_vd.h>
```

Public Member Functions

- [Pila_max](#) ()
Constructor por defecto.
- [Pila_max](#) (const [Pila_max](#) &pm)
Constructor de copias.
- [~Pila_max](#) ()
Destructor.
- void [clear](#) ()
Vacia la pila.
- [Pila_max](#) & [operator=](#) (const [Pila_max](#) &pm)
Operador de asignacion.
- void [push](#) (int e)
Aniade un elemento "encima" del tope de la pila.
- void [pop](#) ()

- Quita el elemento del tope de la pila.*
 - `elemento peek () const`
Devuelve el elemento del tope de la pila y el maximo.
- `int elem () const`
Devuelve el elemento del tope de la pila.
- `int max () const`
Devuelve el maximo de la pila.
- `bool empty () const`
Comprueba si la pila esta vacia.
- `int size () const`
Devuelve el numero de elementos de la pila.
- `bool operator== (const Pila_max &pm)`
Operador de igualdad. Dos pilas son iguales si tienen los mismos maximos.
- `bool operator!= (const Pila_max &pm)`
Operador de desigualdad. Dos pilas son distintas si tienen maximos distintos.
- `bool operator< (const Pila_max &pm)`
Operador de comparacion menor que. Una pila es menor que otra si su maximo es menor.
- `bool operator> (const Pila_max &pm)`
Operador de comparacion mayor que. Una pila es mayor que otra si su maximo es mayor.

Friends

- `ostream & operator<< (ostream &os, const Pila_max &pm)`
Operador flujo de salida.
- `istream & operator>> (istream &is, Pila_max &pm)`
Operador flujo de entrada.

3.2.1 Detailed Description

T.D.A. [Pila_max](#).

Una instancia v del tipo de datos abstracto [Pila_max](#) sobre el tipo `entero` es una lista de pares de elementos $\langle \text{entero}, \text{entero} \rangle$ con un funcionamiento *LIFO* (Last In, First Out). En una pila, las operaciones de insercion y borrado de elementos tienen lugar en uno de los extremos denominado *Tope*. Una [pila_max](#) de longitud n la denotamos

- $[\langle a_1, \max\{a_1 \dots a_n\} \rangle, \langle a_2, \max\{a_2 \dots a_n\} \rangle, \langle a_3, \max\{a_3 \dots a_n\} \rangle, \dots, \langle a_n, a_n \rangle]$

donde a_i es el elemento de la posición i -ésima. Y $\max\{a_i, \dots, a_j\}$ es el máximo del conjunto de $*$ elementos que van desde a_i a a_j . En esta pila, tendremos acceso únicamente al elemento del *Tope*, es decir, a a_n . El borrado o consulta de un elemento será sobre a_n , y la inserción de un nuevo elemento se hará sobre este. Quedando el elemento insertado como *Tope* de la pila.

Si $n=0$ diremos que la pila está vacía.

El espacio requerido para el almacenamiento es $O(n)$. Donde n es el número de elementos de la [Pila_max](#).

Author

Ana Isabel Guerrero Tejera
 Julian Arenas Guerrero

Date

Noviembre 2017

3.2.2 Constructor & Destructor Documentation

3.2.2.1 Pila_max::Pila_max (const Pila_max & *pm*)

Constructor de copias.

Parameters

<i>pm</i>	La pila de la que se hara la copia.
-----------	-------------------------------------

3.2.3 Member Function Documentation

3.2.3.1 bool Pila_max::operator!= (const Pila_max & *pm*) [inline]

Operador de desigualdad. Dos pilas son distintas si tienen maximos distintos.

Parameters

<i>pm</i>	La pila con la que comparar
-----------	-----------------------------

3.2.3.2 bool Pila_max::operator< (const Pila_max & *pm*) [inline]

Operador de comparacion menor que. Una pila es menor que otra si su maximo es menor.

Parameters

<i>pm</i>	La pila con la que comparar
-----------	-----------------------------

3.2.3.3 Pila_max & Pila_max::operator= (const Pila_max & *pm*)

Operador de asignacion.

Parameters

<i>pm</i>	La pila que se va a asignar.
-----------	------------------------------

3.2.3.4 bool Pila_max::operator== (const Pila_max & *pm*) [inline]

Operador de igualdad. Dos pilas son iguales si tienen los mismos maximos.

Parameters

<i>pm</i>	La pila con la que comparar
-----------	-----------------------------

3.2.3.5 `bool Pila_max::operator> (const Pila_max & pm)` `[inline]`

Operador de comparacion mayor que. Una pila es mayor que otra si su maximo es mayor.

Parameters

<i>pm</i>	La pila con la que comparar
-----------	-----------------------------

3.2.3.6 `void Pila_max::push (int e)`

Aniade un elemento "encima" del tope de la pila.

Parameters

<i>e</i>	Elemento que se va a aniadir.
----------	-------------------------------

The documentation for this class was generated from the following files:

- [pila_max_vd.h](#)
- [pila_max_vd.cpp](#)

Chapter 4

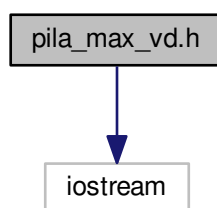
File Documentation

4.1 pila_max_vd.h File Reference

Fichero cabecera del TDA Pila max.

```
#include <iostream>
```

Include dependency graph for pila_max_vd.h:



Classes

- struct [elemento](#)
- class [Pila_max](#)
T.D.A. [Pila_max](#).

Functions

- ostream & **operator**<< (ostream &os, const [elemento](#) &e)

4.1.1 Detailed Description

Fichero cabecera del TDA Pila max.

Gestiona una secuencia de elementos con facilidades para la insercion y borrado de elementos en un extremo

Index

elemento, [5](#)

operator!=
 Pila_max, [7](#)

operator<
 Pila_max, [7](#)

operator>
 Pila_max, [8](#)

operator=
 Pila_max, [7](#)

operator==
 Pila_max, [7](#)

Pila_max, [5](#)
 operator!=, [7](#)
 operator<, [7](#)
 operator>, [8](#)
 operator=, [7](#)
 operator==, [7](#)
 Pila_max, [7](#)
 push, [8](#)

pila_max_vd.h, [9](#)

push
 Pila_max, [8](#)