

JavaScript is disabled on your browser.

[PACKAGE](#) [CLASS](#) [USE](#) [TREE](#) [DEPRECATED](#) [INDEX](#) [HELP](#)

[SUMMARY: NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

SEARCH:

Class VectorHeap<E extends java.lang.Comparable<E>>

java.lang.Object
VectorHeap<E>

Type Parameters:

E -

All Implemented Interfaces:

[interfacePriorityQueue<E>](#)

```
public class VectorHeap<E extends java.lang.Comparable<E>>
    extends java.lang.Object
    implements interfacePriorityQueue<E>
```

Author:

oscar IMPLEMENTACION DE VECTOR

Field Summary

Fields

Modifier and Type	Field	Description
private java.util.Vector< E >	info	ASIGNACION DE NOMBRE AL VECTOR

Constructor Summary

Constructors

Constructor	Description
VectorHeap ()	CONSTRUCTOR DEL METODO VECTOR HEAP

Constructor**Description**

PACKAGE: [VectorHeap](#) CLASS: [USE TREE](#) DEPRECATED: [INDEX](#) [CONSTRUYE UNA NUEVA QUEUE](#)
 (java.util.Vector<[E](#)> v)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) — DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

SEARCH:

Method Summary

All Methods **Static Methods** **Instance Methods** **Concrete Methods**

Modifier and Type	Method	Description
void	add (E value)	AGREGA UN VALOR AL VECTOR
void	clear ()	LIMPIA EL VECTOR
private static int	derecha (int i)	ENTERO DEL INDICE DERECHO
E	getFirst ()	TOMA EL PRIMER VALOR DEL VECTOR
boolean	isEmpty ()	INDICA SI EL VECTOR ESTA VACIO
private static int	izquierda (int i)	ENTERO DEL INDICE IZQUIERDO
private static int	origen (int i)	
private void	percolateUP (int escalon)	MUEVE EL NODO A SU NUEVA UBICACION
private void	Pushdownroot (int raiz)	MUEVE EL NODO DESDE LA RAIZ A UNA POSICION ADECUADA
E	remove ()	QUITA UN VALOR DEL VECTOR
int	size ()	DETERMINA EL TAMAÑO DEL VECTOR
java.lang.String	toString ()	CONVIERTE LOS ELEMENTOS EN STRING

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

[PACKAGE](#) [CLASS](#) [USE](#) [TREE](#) [DEPRECATED](#) [INDEX](#) [HELP](#)

Field Details

[SUMMARY: NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

info

SEARCH:

```
private java.util.Vector<E> extends java.lang.Comparable<E>> info
```

ASIGNACION DE NOMBRE AL VECTOR

Constructor Details

VectorHeap

```
public VectorHeap()
```

CONSTRUCTOR DEL METODO VECTOR HEAP

VectorHeap

```
public VectorHeap(java.util.Vector<E> v)
```

CONSTRUYE UNA NUEVA QUEUE

Parameters:

v -

Method Details

getFirst

```
public E getFirst()
```

TOMA EL PRIMER VALOR DEL VECTOR

Specified by:

```
getFirst in interface interfacePriorityQueue<E extends  
java.lang.Comparable<E>>
```

PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

Returns:

SUMMARY: NESTED | [FIELD](#) | [CONSTR](#) | [METHOD](#) [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

SEARCH:

remove

```
public E remove()
```

QUITA UN VALOR DEL VECTOR

Specified by:

```
remove in interface interfacePriorityQueue<E extends  
java.lang.Comparable<E>>
```

Returns:**add**

```
public void add(E value)
```

AGREGA UN VALOR AL VECTOR

Specified by:

```
add in interface interfacePriorityQueue<E extends java.lang.Comparable<E>>
```

Parameters:

value -

isEmpty

```
public boolean isEmpty()
```

INDICA SI EL VECTOR ESTA VACIO

Specified by:

```
isEmpty in interface interfacePriorityQueue<E extends  
java.lang.Comparable<E>>
```

Returns:**size**

```
public int size()
```

DETERMINA EL TAMAÑO DEL VECTOR

PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

Specified by:

size in interface interfacePriorityQueue<E extends java.lang.Comparable<E>>
SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Returns:

SEARCH:

clear

```
public void clear()
```

LIMPIA EL VECTOR

Specified by:

clear in interface interfacePriorityQueue<E extends
java.lang.Comparable<E>>

percolateUP

```
private void percolateUP(int escalon)
```

MUEVE EL NODO A SU NUEVA UBICACION

Parameters:

escalon -

Pushdownroot

```
private void Pushdownroot(int raiz)
```

MUEVE EL NODO DESDE LA RAIZ A UNA POSICION ADECUADA

Parameters:

raiz -

origen

```
private static int origen(int i)
```

Parameters:

i -

Returns:

[PACKAGE](#) [CLASS](#) [USE](#) [TREE](#) [DEPRECATED](#) [INDEX](#) [HELP](#)

izquierda

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

```
private static int izquierda(int i)
```

SEARCH:

ENTERO DEL INDICE IZQUIERDO

Parameters:

i -

Returns:

derecha

```
private static int derecha(int i)
```

ENTERO DEL INDICE DERECHO

Parameters:

i -

Returns:

toString

```
public java.lang.String toString()
```

CONVIERTE LOS ELEMENTOS EN STRING

Overrides:

toString in class java.lang.Object

Returns:

[PACKAGE](#) [CLASS](#) [USE](#) [TREE](#) [DEPRECATED](#) [INDEX](#) [HELP](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)