

## Practica 6 Jose Antonio Ruiz Millan

Generated by Doxygen 1.8.11



# Contents

<b>1</b>	<b>Class Index</b>	<b>1</b>
1.1	Class List . . . . .	1
<b>2</b>	<b>File Index</b>	<b>3</b>
2.1	File List . . . . .	3
<b>3</b>	<b>Class Documentation</b>	<b>5</b>
3.1	clinvar Class Reference . . . . .	5
3.1.1	Member Typedef Documentation . . . . .	6
3.1.1.1	enfermedad_iterator . . . . .	6
3.1.2	Constructor & Destructor Documentation . . . . .	6
3.1.2.1	clinvar() . . . . .	6
3.1.3	Member Function Documentation . . . . .	6
3.1.3.1	begin() . . . . .	6
3.1.3.2	ebegin() . . . . .	6
3.1.3.3	ehend() . . . . .	7
3.1.3.4	end() . . . . .	7
3.1.3.5	erase(IDmut ID) . . . . .	7
3.1.3.6	find_Enf(IDenf ID) . . . . .	7
3.1.3.7	find_Mut(IDmut ID) . . . . .	8
3.1.3.8	gbegin() . . . . .	8
3.1.3.9	gend() . . . . .	8
3.1.3.10	getEnfermedades(mutacion &mut) . . . . .	8
3.1.3.11	getEnfermedades(string keyword) . . . . .	8

3.1.3.12	<a href="#">getMutacionesEnf(IDenf ID)</a>	9
3.1.3.13	<a href="#">getMutacionesGen(IDgen ID)</a>	9
3.1.3.14	<a href="#">insert(const mutacion &amp;x)</a>	9
3.1.3.15	<a href="#">load(string nombreDB)</a>	10
3.1.3.16	<a href="#">lower_bound(string cromosoma, unsigned int posicion)</a>	10
3.1.3.17	<a href="#">topKMutaciones(int k, string keyword)</a>	10
3.1.3.18	<a href="#">upper_bound(string cromosoma, unsigned int posicion)</a>	10
3.2	<a href="#">clinvar::gen_iterator Class Reference</a>	11
3.2.1	<a href="#">Constructor &amp; Destructor Documentation</a>	11
3.2.1.1	<a href="#">gen_iterator()</a>	11
3.2.1.2	<a href="#">gen_iterator(const gen_iterator &amp;x)</a>	11
3.2.2	<a href="#">Member Function Documentation</a>	12
3.2.2.1	<a href="#">operator!=(const gen_iterator &amp;x) const</a>	12
3.2.2.2	<a href="#">operator*()</a>	12
3.2.2.3	<a href="#">operator++()</a>	12
3.2.2.4	<a href="#">operator--()</a>	12
3.2.2.5	<a href="#">operator=(const gen_iterator &amp;x)</a>	12
3.2.2.6	<a href="#">operator==(const gen_iterator &amp;x) const</a>	13
3.2.3	<a href="#">Friends And Related Function Documentation</a>	13
3.2.3.1	<a href="#">clinvar</a>	13
3.3	<a href="#">clinvar::iterator Class Reference</a>	13
3.3.1	<a href="#">Constructor &amp; Destructor Documentation</a>	14
3.3.1.1	<a href="#">iterator()</a>	14
3.3.1.2	<a href="#">iterator(const iterator &amp;x)</a>	14
3.3.2	<a href="#">Member Function Documentation</a>	14
3.3.2.1	<a href="#">operator!=(const iterator &amp;x) const</a>	14
3.3.2.2	<a href="#">operator*()</a>	14
3.3.2.3	<a href="#">operator++()</a>	15
3.3.2.4	<a href="#">operator--()</a>	15
3.3.2.5	<a href="#">operator=(const iterator &amp;x)</a>	15
3.3.2.6	<a href="#">operator=(const unordered_map&lt; IDmut, set&lt; mutacion &gt;::iterator &gt;::iterator &amp;x)</a>	15
3.3.2.7	<a href="#">operator==(const iterator &amp;x) const</a>	16
3.3.3	<a href="#">Friends And Related Function Documentation</a>	16
3.3.3.1	<a href="#">clinvar</a>	16
3.4	<a href="#">ProbMutaciones Class Reference</a>	16
3.4.1	<a href="#">Detailed Description</a>	16
3.4.2	<a href="#">Member Function Documentation</a>	16
3.4.2.1	<a href="#">operator()(const mutacion &amp;m1, const mutacion &amp;m2)</a>	16

<b>CONTENTS</b>	<b>v</b>
<b>4 File Documentation</b>	<b>17</b>
4.1 include/clinvar.h File Reference . . . . .	17
4.2 src/principal.cpp File Reference . . . . .	17
4.2.1 Function Documentation . . . . .	17
4.2.1.1 main() . . . . .	17
<b>Index</b>	<b>19</b>



# Chapter 1

## Class Index

### 1.1 Class List

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

<a href="#">clinvar</a> . . . . .	5
<a href="#">clinvar::gen_iterator</a> . . . . .	11
<a href="#">clinvar::iterator</a> . . . . .	13
<a href="#">ProbMutaciones</a>	
Funtor que utilizaremos para ordenar por probabilidad . . . . .	16





## Chapter 2

# File Index

### 2.1 File List

Here is a list of all files with brief descriptions:

include/ <a href="#">clinvar.h</a> . . . . .	17
src/ <a href="#">principal.cpp</a> . . . . .	17



## Chapter 3

# Class Documentation

### 3.1 clinvar Class Reference

```
#include <clinvar.h>
```

#### Classes

- class [gen\\_iterator](#)
- class [iterator](#)

#### Public Types

- typedef map< [IDenf](#), enfermedad >::iterator [enfermedad\\_iterator](#)

#### Public Member Functions

- [clinvar](#) ()
- void [load](#) (string nombreDB)  
*Se encarga de leer los elementos de un fichero dado por el argumento nombreDB, e insertar toda la información en ClinVar.*
- void [insert](#) (const mutacion &x)  
*Este método se encarga de insertar una nueva mutación en ClinVar. Esto implica actualizar todas las estructuras necesarias para mantener la coherencia interna de la representación propuesta.*
- bool [erase](#) ([IDmut](#) ID)  
*En este caso, se trata de borrar una mutación de la base de datos dado su ID. No sólo borra la mutación del repositorio principal de datos sino que además se encarga de borrar toda referencia a dicho elemento dentro de él. En el caso de que una enfermedad estuviese asociada únicamente a la mutación que está siendo eliminada, esta enfermedad también debe eliminarse de ClinVar.*
- [clinvar::iterator](#) [find\\_Mut](#) ([IDmut](#) ID)  
*Busca la mutación con identificador ID dentro de ClinVar.*
- [clinvar::enfermedad\\_iterator](#) [find\\_Enf](#) ([IDenf](#) ID)  
*Busca la enfermedad con identificador ID dentro de ClinVar.*
- vector< enfermedad > [getEnfermedades](#) (mutacion &mut)  
*Busca todas las enfermedades asociadas a una mutación en la base de datos clinvar.*
- list< [IDenf](#) > [getEnfermedades](#) (string keyword)

- Busca los identificadores de enfermedad que contienen la palabra keyword como parte del nombre de la enfermedad.*

  - `set< IDmut > getMutacionesEnf (IDenf ID)`

*Busca todas las mutaciones que se encuentran asociadas a la enfermedad con identificador ID.*

  - `set< IDmut > getMutacionesGen (IDgen ID)`

*Busca las mutaciones que se encuentran asociadas a un gen determinado dado por ID.*

  - `set< mutacion, ProbMutaciones > topKMutaciones (int k, string keyword)`

*Dado un string "keyword", el sistema recupera todas las enfermedades cuyo nombre contiene keyword.*

  - `iterator lower_bound` (string cromosoma, unsigned int posicion)

*hace la búsqueda por rango considerando el par de valores cromosoma/posición.*

  - `iterator upper_bound` (string cromosoma, unsigned int posicion)

*hace la búsqueda por rango considerando el par de valores cromosoma/posición.*

  - `iterator begin ()`

*begin del conjunto*

  - `iterator end ()`

*end del conjunto.*

  - `enfermedad_iterator ebegin ()`

*begin de enfermedad del conjunto*

  - `enfermedad_iterator eend ()`

*end de enfermedad del conjunto.*

  - `gen_iterator gbegin ()`

*begin de enfermedad del conjunto*

  - `gen_iterator gend ()`

*end de gen del conjunto.*

### 3.1.1 Member Typedef Documentation

3.1.1.1 `typedef map<IDenf, enfermedad>::iterator clinvar::enfermedad_iterator`

### 3.1.2 Constructor & Destructor Documentation

3.1.2.1 `clinvar::clinvar ( )` [inline]

### 3.1.3 Member Function Documentation

3.1.3.1 `iterator clinvar::begin ( )`

begin del conjunto

#### Returns

el iterador correspondiente a la primera mutación según el criterio de ordenación de cada tipo de elemento.

3.1.3.2 `enfermedad_iterator clinvar::ebegin ( )`

begin de enfermedad del conjunto

#### Returns

el iterador correspondiente a la primera enfermedad según el criterio de ordenación de cada tipo de elemento.

### 3.1.3.3 enfermedad\_iterator clinvar::eend ( )

end de enfermedad del conjunto.

#### Returns

el iterador que apunta al elemento siguiente al último elemento según el criterio de ordenación de cada tipo de elemento.

### 3.1.3.4 iterator clinvar::end ( )

end del conjunto.

#### Returns

el iterador que apunta al elemento siguiente al último elemento según el criterio de ordenación de cada tipo de elemento.

### 3.1.3.5 bool clinvar::erase ( IDmut ID )

En este caso, se trata de borrar una mutación de la base de datos dado su ID. No sólo borra la mutación del repositorio principal de datos sino que además se encarga de borrar toda referencia a dicho elemento dentro de él. En el caso de que una enfermedad estuviese asociada únicamente a la mutación que está siendo eliminada, esta enfermedad también debe eliminarse de ClinVar.

#### Parameters

ID	ID de la mutacion ha eliminarse
----	---------------------------------

#### Returns

Devuelve verdadero si el elemento ha sido borrado correctamente, falso en caso contrario.

### 3.1.3.6 clinvar::enfermedad\_iterator clinvar::find\_Enf ( IDenf ID )

Busca la enfermedad con identificador ID dentro de ClinVar.

#### Parameters

ID	ID de la enfermedad ha buscar
----	-------------------------------

#### Returns

iterador que apunta a la enfermedad, si no lo encuentra devuelve [end\(\)](#)

### 3.1.3.7 `clinvar::iterator clinvar::find_Mut ( IDmut ID )`

Busca la mutación con identificador ID dentro de ClinVar.

#### Parameters

<i>ID</i>	ID de la mutacion ha buscar
-----------	-----------------------------

#### Returns

El iterador a la mutacion, en caso de no existir, devuelve `end()`

### 3.1.3.8 `gen_iterator clinvar::gbegin ( )`

begin de enfermedad del conjunto

#### Returns

el iterador correspondiente al primer gen según el criterio de ordenación de cada tipo de elemento.

### 3.1.3.9 `gen_iterator clinvar::gend ( )`

end de gen del conjunto.

#### Returns

el iterador que apunta al elemento siguiente al último elemento según el criterio de ordenación de cada tipo de elemento.

### 3.1.3.10 `vector<enfermedad> clinvar::getEnfermedades ( mutacion & mut )`

Busca todas las enfermedades asociadas a una mutación en la base de datos clinvar.

#### Parameters

<i>mut</i>	mutacion ha buscar para encontrar sus enfermedades asociadas
------------	--

#### Returns

un vector con todas las enfermedades asociadas a una mutación

### 3.1.3.11 `list<IDenf> clinvar::getEnfermedades ( string keyword )`

Busca los identificadores de enfermedad que contienen la palabra keyword como parte del nombre de la enfermedad.

## Parameters

<i>keyword</i>	string que contiene la parte del nombre de la enfermedad ha buscar.
----------------	---

## Returns

una lista de los identificadores de enfermedad que contienen la palabra keyword como parte del nombre de la enfermedad

**3.1.3.12** `set<IDmut> clinvar::getMutacionesEnf ( IDenf ID )`

Busca todas las mutaciones que se encuentran asociadas a la enfermedad con identificador ID.

## Parameters

<i>ID</i>	ID de la enfermedad ha buscar.
-----------	--------------------------------

## Returns

un conjunto ordenado (en orden creciente de IDmut) de todas las mutaciones que se encuentran asociadas a la enfermedad con identificador ID. Si no tuviese ninguna enfermedad asociada, devuelve el conjunto vacío.

**3.1.3.13** `set<IDmut> clinvar::getMutacionesGen ( IDgen ID )`

Busca las mutaciones que se encuentran asociadas a un gen determinado dado por ID.

## Parameters

<i>ID</i>	ID del gen ha buscar.
-----------	-----------------------

## Returns

Devuelve un conjunto de todas las mutaciones que se encuentran asociadas a un gen determinado dado por ID. Si no tuviese ninguno, devuelve el conjunto vacío.

**3.1.3.14** `void clinvar::insert ( const mutacion & x )`

Este método se encarga de insertar una nueva mutación en ClinVar. Esto implica actualizar todas las estructuras necesarias para mantener la coherencia interna de la representación propuesta.

## Parameters

<i>x</i>	mutacion ha insertar en ClinVar
----------	---------------------------------

### 3.1.3.15 void clinvar::load ( string *nombreDB* )

Se encarga de leer los elementos de un fichero dado por el argumento *nombreDB*, e insertar toda la información en ClinVar.

#### Parameters

<i>nombreDB</i>	Nombre del fichero donde se almacenan los datos
-----------------	---

### 3.1.3.16 iterator clinvar::lower\_bound ( string *cromosoma*, unsigned int *posicion* )

hace la búsqueda por rango considerando el par de valores cromosoma/posición.

#### Parameters

<i>cromosoma</i>	string que contiene el cromosoma ha buscar.
<i>posicion</i>	entero positivo que contiene el valor de la posicion.

#### Returns

el iterador que apunta a la primera mutación que es mayor o igual a los parámetros dados en la entrada, si no hay ninguna devuelve end.

### 3.1.3.17 set<mutacion,ProbMutaciones> clinvar::topKMutaciones ( int *k*, string *keyword* )

Dado un string 'keyword', el sistema recupera todas las enfermedades cuyo nombre contiene keyword.

#### Parameters

<i>k</i>	numero de mutaciones ha devolver.
<i>keyword</i>	string con el nombre de la enfermedad

#### Returns

un set ordenado de mutaciones, en orden decreciente de probabilidad, con las k mutaciones más frecuentes en la población asociadas con esas enfermedades.

### 3.1.3.18 iterator clinvar::upper\_bound ( string *cromosoma*, unsigned int *posicion* )

hace la búsqueda por rango considerando el par de valores cromosoma/posición.

#### Parameters

<i>cromosoma</i>	string que contiene el cromosoma ha buscar.
<i>posicion</i>	entero positivo que contiene el valor de la posicion.



### Returns

el iterador que apunta a la primera mutación que es estrictamente mayor a los parámetros dados en la entrada, si no hay ninguna devuelve end.

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

- include/[clinvar.h](#)

## 3.2 clinvar::gen\_iterator Class Reference

```
#include <clinvar.h>
```

### Public Member Functions

- [gen\\_iterator](#) ()  
*Constructor por defecto.*
- [gen\\_iterator](#) (const [gen\\_iterator](#) &x)  
*Constructor de copia.*
- const mutacion & [operator\\*](#) ()  
*Devuelve el contenido al que apunta el iterador.*
- [gen\\_iterator](#) & [operator++](#) ()  
*Adelanta una posicion el iterador.*
- [gen\\_iterator](#) & [operator--](#) ()  
*Atrasa el iterador una posicion.*
- bool [operator==](#) (const [gen\\_iterator](#) &x) const  
*Compara si dos iteradores apuntan al mismo sitio.*
- bool [operator!=](#) (const [gen\\_iterator](#) &x) const  
*Compara si dos iteradores no apuntan al mismo elemento.*
- [gen\\_iterator](#) & [operator=](#) (const [gen\\_iterator](#) &x)  
*Iguala dos iteradores.*

### Friends

- class [clinvar](#)

### 3.2.1 Constructor & Destructor Documentation

#### 3.2.1.1 clinvar::gen\_iterator::gen\_iterator ( ) [inline]

Constructor por defecto.

#### 3.2.1.2 clinvar::gen\_iterator::gen\_iterator ( const gen\_iterator & x ) [inline]

Constructor de copia.

**Parameters**

x	iterador para copiar su contenido
---	-----------------------------------

**3.2.2 Member Function Documentation**

**3.2.2.1** `bool clinvar::gen_iterator::operator!=( const gen_iterator & x ) const` `[inline]`

Compara si dos iteradores no apuntan al mismo elemento.

**Parameters**

x	iterador a comparar con el actual
---	-----------------------------------

**Returns**

true si los iteradores son distintos, false en caso contrario

**3.2.2.2** `const mutacion& clinvar::gen_iterator::operator*( )` `[inline]`

Devuelve el contenido al que apunta el iterador.

**Returns**

elemto del vector al que apunta el iterador

**3.2.2.3** `gen_iterator& clinvar::gen_iterator::operator++( )` `[inline]`

Adelanta una posicion el iterador.

**Returns**

la referencia al propio iterador

**3.2.2.4** `gen_iterator& clinvar::gen_iterator::operator--( )` `[inline]`

Atrasa el iterador una posicion.

**Returns**

la referencia al propio iterador

**3.2.2.5** `gen_iterator& clinvar::gen_iterator::operator=( const gen_iterator & x )` `[inline]`

Iguala dos iteradores.

## Parameters

x	iterador del que copiaremos la informacion
---	--

## Returns

la referencia al propio iterador

**3.2.2.6** `bool clinvar::gen_iterator::operator==( const gen_iterator & x ) const` `[inline]`

Compara si dos iteradores apuntan al mismo sitio.

## Parameters

x	iterador a comparar con el actual
---	-----------------------------------

## Returns

true si son iguales o false en caso contrario

### 3.2.3 Friends And Related Function Documentation

**3.2.3.1** `friend class clinvar` `[friend]`

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

- [include/clinvar.h](#)

## 3.3 clinvar::iterator Class Reference

```
#include <clinvar.h>
```

### Public Member Functions

- [iterator](#) ()  
*Constructor por defecto.*
- [iterator](#) (const [iterator](#) &x)  
*Constructor de copia.*
- const mutacion & [operator\\*](#) ()  
*Devuelve el contenido al que apunta el iterador.*
- [iterator](#) & [operator++](#) ()  
*Adelanta una posicion el iterador.*
- [iterator](#) & [operator--](#) ()  
*Atrasa el iterador una posicion.*

- `bool operator== (const iterator &x) const`  
*Compara si dos iteradores apuntan al mismo sitio.*
- `bool operator!= (const iterator &x) const`  
*Compara si dos iteradores no apuntan al mismo elemento.*
- `iterator & operator= (const iterator &x)`  
*Iguala dos iteradores.*
- `iterator & operator= (const unordered_map< IDmut, set< mutacion >::iterator >::iterator &x)`  
*Transforma un iterador de IDm\_map ha iterator.*

## Friends

- class `clinvar`

### 3.3.1 Constructor & Destructor Documentation

#### 3.3.1.1 `clinvar::iterator::iterator ( )` `[inline]`

Constructor por defecto.

#### 3.3.1.2 `clinvar::iterator::iterator ( const iterator & x )` `[inline]`

Constructor de copia.

##### Parameters

<code>x</code>	iterador para copiar su contenido
----------------	-----------------------------------

### 3.3.2 Member Function Documentation

#### 3.3.2.1 `bool clinvar::iterator::operator!= ( const iterator & x ) const` `[inline]`

Compara si dos iteradores no apuntan al mismo elemento.

##### Parameters

<code>x</code>	iterador a comparar con el actual
----------------	-----------------------------------

##### Returns

true si los iteradores son distintos, false en caso contrario

#### 3.3.2.2 `const mutacion& clinvar::iterator::operator* ( )` `[inline]`

Devuelve el contenido al que apunta el iterador.

**Returns**

elemto del vector al que apunta el iterador

**3.3.2.3** `iterator& clinvar::iterator::operator++ ( ) [inline]`

Adelanta una posicion el iterador.

**Returns**

la referencia al propio iterador

**3.3.2.4** `iterator& clinvar::iterator::operator-- ( ) [inline]`

Atrasa el iterador una posicion.

**Returns**

la referencia al propio iterador

**3.3.2.5** `iterator& clinvar::iterator::operator= ( const iterator & x ) [inline]`

Iguala dos iteradores.

**Parameters**

<code>x</code>	iterador del que copiaremos la informacion
----------------	--

**Returns**

la referencia al propio iterador

**3.3.2.6** `iterator& clinvar::iterator::operator= ( const unordered_map< IDmut, set< mutacion >::iterator >::iterator & x ) [inline]`

Transforma un iterador de IDm\_map ha iterator.

**Parameters**

<code>x</code>	iterador del que copiaremos la informacion
----------------	--

**Returns**

la referencia al propio iterador

**3.3.2.7** `bool clinvar::iterator::operator==( const iterator & x ) const` `[inline]`

Compara si dos iteradores apuntan al mismo sitio.

#### Parameters

x	iterador a comparar con el actual
---	-----------------------------------

#### Returns

true si son iguales o false en caso contrario

### 3.3.3 Friends And Related Function Documentation

**3.3.3.1** `friend class clinvar` `[friend]`

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

- [include/clinvar.h](#)

## 3.4 ProbMutaciones Class Reference

Funtor que utilizaremos para ordenar por probabilidad.

```
#include <clinvar.h>
```

#### Public Member Functions

- `bool operator()` (const mutacion &m1, const mutacion &m2)

### 3.4.1 Detailed Description

Funtor que utilizaremos para ordenar por probabilidad.

### 3.4.2 Member Function Documentation

**3.4.2.1** `bool ProbMutaciones::operator()( const mutacion & m1, const mutacion & m2 )` `[inline]`

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

- [include/clinvar.h](#)

## Chapter 4

# File Documentation

### 4.1 include/clinvar.h File Reference

```
#include <string>
#include <map>
#include <unordered_map>
#include <set>
#include <vector>
#include <list>
#include "enfermedad.h"
#include "mutacion.h"
Include dependency graph for clinvar.h:
```

### 4.2 src/principal.cpp File Reference

```
#include <iostream>
#include <list>
#include <set>
#include <string>
#include <unistd.h>
#include "clinvar.h"
Include dependency graph for principal.cpp:
```

#### Functions

- int [main](#) ()

#### 4.2.1 Function Documentation

##### 4.2.1.1 int main ( )





# Index

begin  
    clinvar, 6

clinvar, 5  
    begin, 6  
    clinvar, 6  
    clinvar::gen\_iterator, 13  
    clinvar::iterator, 16  
    ebegin, 6  
    eend, 6  
    end, 7  
    enfermedad\_iterator, 6  
    erase, 7  
    find\_Enf, 7  
    find\_Mut, 7  
    gbegin, 8  
    gend, 8  
    getEnfermedades, 8  
    getMutacionesEnf, 9  
    getMutacionesGen, 9  
    insert, 9  
    load, 9  
    lower\_bound, 10  
    topKMutaciones, 10  
    upper\_bound, 10

clinvar::gen\_iterator, 11  
    clinvar, 13  
    gen\_iterator, 11  
    operator!=, 12  
    operator\*, 12  
    operator++, 12  
    operator--, 12  
    operator=, 12  
    operator==, 13

clinvar::iterator, 13  
    clinvar, 16  
    iterator, 14  
    operator!=, 14  
    operator\*, 14  
    operator++, 15  
    operator--, 15  
    operator=, 15  
    operator==, 15

ebegin  
    clinvar, 6

eend  
    clinvar, 6

end  
    clinvar, 7

enfermedad\_iterator  
    clinvar, 6

erase  
    clinvar, 7

find\_Enf  
    clinvar, 7

find\_Mut  
    clinvar, 7

gbegin  
    clinvar, 8

gen\_iterator  
    clinvar::gen\_iterator, 11

gend  
    clinvar, 8

getEnfermedades  
    clinvar, 8

getMutacionesEnf  
    clinvar, 9

getMutacionesGen  
    clinvar, 9

include/clinvar.h, 17

insert  
    clinvar, 9

iterator  
    clinvar::iterator, 14

load  
    clinvar, 9

lower\_bound  
    clinvar, 10

main  
    principal.cpp, 17

operator!=  
    clinvar::gen\_iterator, 12  
    clinvar::iterator, 14

operator\*  
    clinvar::gen\_iterator, 12  
    clinvar::iterator, 14

operator()  
    ProbMutaciones, 16

operator++  
    clinvar::gen\_iterator, 12  
    clinvar::iterator, 15

operator--  
    clinvar::gen\_iterator, 12  
    clinvar::iterator, 15

operator=  
    clinvar::gen\_iterator, [12](#)  
    clinvar::iterator, [15](#)  
operator==  
    clinvar::gen\_iterator, [13](#)  
    clinvar::iterator, [15](#)  
  
principal.cpp  
    main, [17](#)  
ProbMutaciones, [16](#)  
    operator(), [16](#)  
  
src/principal.cpp, [17](#)  
  
topKMutaciones  
    clinvar, [10](#)  
  
upper\_bound  
    clinvar, [10](#)