

# Виртуальное ромбовидное наследование

Программная документация



1 Иерархический список классов	1
1.1 Иерархия классов	1
2 Алфавитный указатель классов	3
2.1 Классы	3
3 Классы	5
3.1 Шаблон класса A< SizeX, SizeY >	5
3.1.1 Конструктор(ы)	6
3.1.2 Методы	7
3.1.3 Документация по друзьям класса и функциям, относящимся к классу	7
3.1.4 Данные класса	7
3.2 Шаблон класса B< SizeX, SizeY >	8
3.2.1 Конструктор(ы)	9
3.2.2 Методы	10
3.2.3 Документация по друзьям класса и функциям, относящимся к классу	10
3.2.4 Данные класса	11
3.3 Шаблон класса C< SizeX, SizeY >	11
3.3.1 Конструктор(ы)	13
3.3.2 Методы	14
3.3.3 Документация по друзьям класса и функциям, относящимся к классу	14
3.3.4 Данные класса	14
3.4 Шаблон класса D< SizeX, SizeY >	15
3.4.1 Конструктор(ы)	16
3.4.2 Методы	17
3.4.3 Документация по друзьям класса и функциям, относящимся к классу	17
3.4.4 Данные класса	18
3.5 Шаблон класса E< SizeX, SizeY >	18
3.5.1 Конструктор(ы)	20
3.5.2 Методы	21
3.5.3 Документация по друзьям класса и функциям, относящимся к классу	21
3.5.4 Данные класса	22
3.6 Шаблон класса F< SizeX, SizeY >	22
3.6.1 Конструктор(ы)	24
3.6.2 Методы	25
3.6.3 Документация по друзьям класса и функциям, относящимся к классу	26
3.6.4 Данные класса	26
3.7 Шаблон класса G< SizeX, SizeY >	26
3.7.1 Конструктор(ы)	29
3.7.2 Методы	29
3.7.3 Документация по друзьям класса и функциям, относящимся к классу	30
3.7.4 Данные класса	30
Предметный указатель	31



## Раздел 1

# Иерархический список классов

### 1.1 Иерархия классов

Иерархия классов.

A< SizeX, SizeY > . . . . .	5
B< SizeX, SizeY > . . . . .	8
C< SizeX, SizeY > . . . . .	11
F< SizeX, SizeY > . . . . .	22
G< SizeX, SizeY > . . . . .	26
D< SizeX, SizeY > . . . . .	15
E< SizeX, SizeY > . . . . .	18
F< SizeX, SizeY > . . . . .	22



## Раздел 2

# Алфавитный указатель классов

### 2.1 Классы

Классы с их кратким описанием.

A< SizeX, SizeY > . . . . .	5
B< SizeX, SizeY > . . . . .	8
C< SizeX, SizeY > . . . . .	11
D< SizeX, SizeY > . . . . .	15
E< SizeX, SizeY > . . . . .	18
F< SizeX, SizeY > . . . . .	22
G< SizeX, SizeY > . . . . .	26





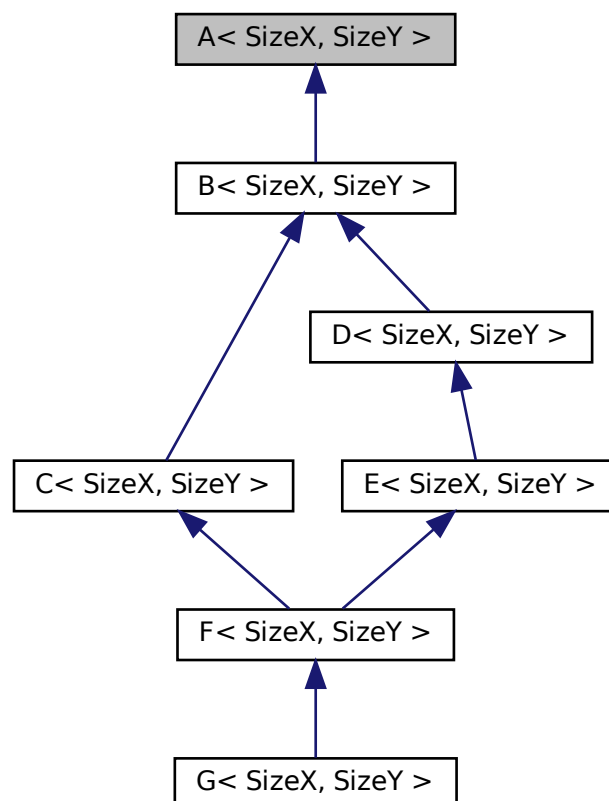
## Раздел 3

# Классы

### 3.1 Шаблон класса A< SizeX, SizeY >

```
#include <classes.h>
```

Граф наследования: A< SizeX, SizeY >:



## Открытые члены

- `A()`
- `A(const A &other)`
- `A & operator= (const A &other)`
- `A(A &&other) noexcept`
- `A & operator= (A &&other) noexcept`
- `virtual ~A()`

## Открытые атрибуты

- `int IntegerA`
- `arma::mat MatrixA`

## Друзья

- `void swap(A< SizeX, SizeY > &lhs, A< SizeX, SizeY > &rhs) noexcept`

### 3.1.1 Конструктор(ы)

#### 3.1.1.1 `A()` [1/3]

```
template<size_t SizeX, size_t SizeY>
A< SizeX, SizeY >::A() [inline]
```

#### 3.1.1.2 `A()` [2/3]

```
template<size_t SizeX, size_t SizeY>
A< SizeX, SizeY >::A (
    const A< SizeX, SizeY > & other ) [inline]
```

#### 3.1.1.3 `A()` [3/3]

```
template<size_t SizeX, size_t SizeY>
A< SizeX, SizeY >::A (
    A< SizeX, SizeY > && other ) [inline], [noexcept]
```

## 3.1.1.4 ~A()

```
template<size_t SizeX, size_t SizeY>
virtual A< SizeX, SizeY >::~~A ( ) [inline], [virtual]
```

## 3.1.2 Методы

## 3.1.2.1 operator=() [1/2]

```
template<size_t SizeX, size_t SizeY>
A & A< SizeX, SizeY >::operator= (
    A< SizeX, SizeY > && other ) [inline], [noexcept]
```

## 3.1.2.2 operator=() [2/2]

```
template<size_t SizeX, size_t SizeY>
A & A< SizeX, SizeY >::operator= (
    const A< SizeX, SizeY > & other ) [inline]
```

## 3.1.3 Документация по друзьям класса и функциям, относящимся к классу

## 3.1.3.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
    A< SizeX, SizeY > & lhs,
    A< SizeX, SizeY > & rhs ) [friend]
```

## 3.1.4 Данные класса

## 3.1.4.1 IntegerA

```
template<size_t SizeX, size_t SizeY>
int A< SizeX, SizeY >::IntegerA
```

### 3.1.4.2 MatrixA

```
template<size_t SizeX, size_t SizeY>  
arma::mat A< SizeX, SizeY >::MatrixA
```

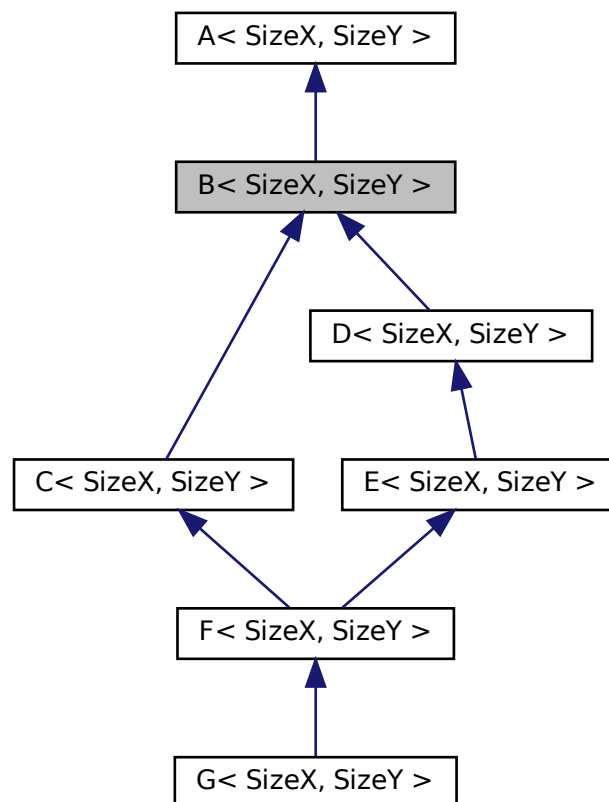
Объявления и описания членов класса находятся в файле:

- classes.h

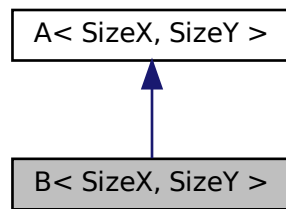
## 3.2 Шаблон класса B< SizeX, SizeY >

```
#include <classes.h>
```

Граф наследования: B< SizeX, SizeY >:



Граф связей класса B< SizeX, SizeY >:



### Открытые члены

- B ()
- B (const B &other)
- B & operator= (const B &other)
- B (B &&other)
- B & operator= (B &&other)
- virtual ~B ()

### Открытые атрибуты

- int IntegerB
- arma::mat MatrixB

### Друзья

- void swap (B< SizeX, SizeY > &lhs, B< SizeX, SizeY > &rhs)

## 3.2.1 Конструктор(ы)

### 3.2.1.1 B() [1/3]

```

template<size_t SizeX, size_t SizeY>
B< SizeX, SizeY >::B ( ) [inline]
  
```

### 3.2.1.2 B() [2/3]

```
template<size_t SizeX, size_t SizeY>
B< SizeX, SizeY >::B (
    const B< SizeX, SizeY > & other )    [inline]
```

### 3.2.1.3 B() [3/3]

```
template<size_t SizeX, size_t SizeY>
B< SizeX, SizeY >::B (
    B< SizeX, SizeY > && other )    [inline]
```

### 3.2.1.4 ~B()

```
template<size_t SizeX, size_t SizeY>
virtual B< SizeX, SizeY >::~~B ( )    [inline], [virtual]
```

## 3.2.2 Методы

### 3.2.2.1 operator=() [1/2]

```
template<size_t SizeX, size_t SizeY>
B & B< SizeX, SizeY >::operator= (
    B< SizeX, SizeY > && other )    [inline]
```

### 3.2.2.2 operator=() [2/2]

```
template<size_t SizeX, size_t SizeY>
B & B< SizeX, SizeY >::operator= (
    const B< SizeX, SizeY > & other )    [inline]
```

## 3.2.3 Документация по друзьям класса и функциям, относящимся к классу

### 3.2.3.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
    B< SizeX, SizeY > & lhs,
    B< SizeX, SizeY > & rhs ) [friend]
```

### 3.2.4 Данные класса

#### 3.2.4.1 IntegerB

```
template<size_t SizeX, size_t SizeY>
int B< SizeX, SizeY >::IntegerB
```

#### 3.2.4.2 MatrixB

```
template<size_t SizeX, size_t SizeY>
arma::mat B< SizeX, SizeY >::MatrixB
```

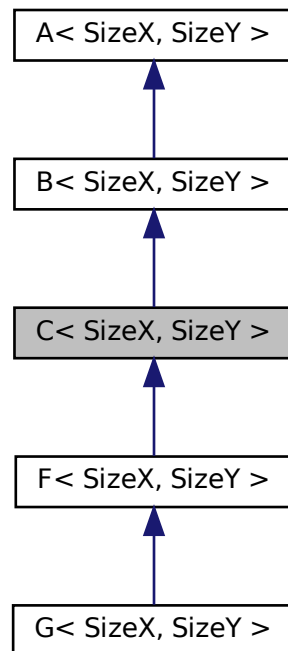
Объявления и описания членов класса находятся в файле:

- classes.h

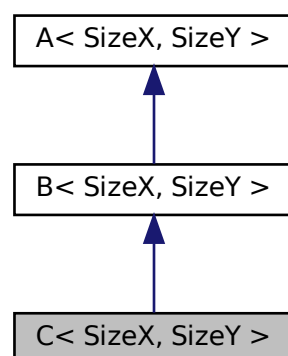
## 3.3 Шаблон класса C< SizeX, SizeY >

```
#include <classes.h>
```

Граф наследования: C < SizeX, SizeY >:



Граф связей класса C < SizeX, SizeY >:



Открытые члены

- `C()`



- C (const C &other)
- C & operator= (const C &other)
- C (C &&other) noexcept
- C & operator= (C &&other) noexcept
- virtual ~C ()

### Открытые атрибуты

- int IntegerC
- arma::mat MatrixC

### Друзья

- void swap (C< SizeX, SizeY > &lhs, C< SizeX, SizeY > &rhs) noexcept

## 3.3.1 Конструктор(ы)

### 3.3.1.1 C() [1/3]

```
template<size_t SizeX, size_t SizeY>
C< SizeX, SizeY >::C ( ) [inline]
```

### 3.3.1.2 C() [2/3]

```
template<size_t SizeX, size_t SizeY>
C< SizeX, SizeY >::C (
    const C< SizeX, SizeY > & other ) [inline]
```

### 3.3.1.3 C() [3/3]

```
template<size_t SizeX, size_t SizeY>
C< SizeX, SizeY >::C (
    C< SizeX, SizeY > && other ) [inline], [noexcept]
```

### 3.3.1.4 ~C()

```
template<size_t SizeX, size_t SizeY>
virtual C< SizeX, SizeY >::~C ( ) [inline], [virtual]
```

### 3.3.2 Методы

#### 3.3.2.1 operator=() [1/2]

```
template<size_t SizeX, size_t SizeY>
C & C< SizeX, SizeY >::operator= (
    C< SizeX, SizeY > && other ) [inline], [noexcept]
```

#### 3.3.2.2 operator=() [2/2]

```
template<size_t SizeX, size_t SizeY>
C & C< SizeX, SizeY >::operator= (
    const C< SizeX, SizeY > & other ) [inline]
```

### 3.3.3 Документация по друзьям класса и функциям, относящимся к классу

#### 3.3.3.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
    C< SizeX, SizeY > & lhs,
    C< SizeX, SizeY > & rhs ) [friend]
```

### 3.3.4 Данные класса

#### 3.3.4.1 IntegerC

```
template<size_t SizeX, size_t SizeY>
int C< SizeX, SizeY >::IntegerC
```

#### 3.3.4.2 MatrixC

```
template<size_t SizeX, size_t SizeY>
arma::mat C< SizeX, SizeY >::MatrixC
```

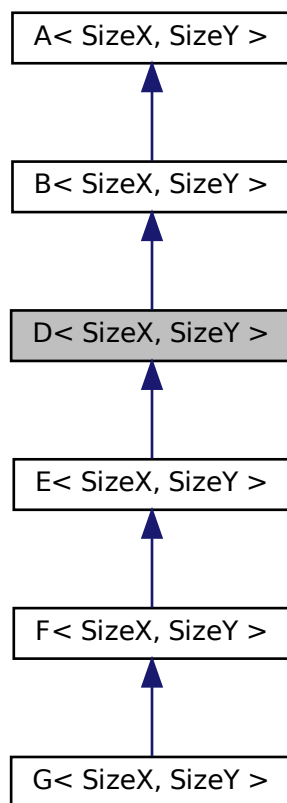
Объявления и описания членов класса находятся в файле:

- classes.h

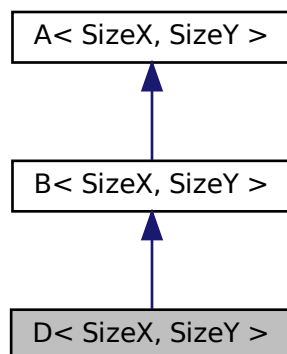
### 3.4 Шаблон класса D< SizeX, SizeY >

```
#include <classes.h>
```

Граф наследования:D< SizeX, SizeY >:



Граф связей класса `D< SizeX, SizeY >`:



### Открытые члены

- `D()`
- `D(const D &other)`
- `D & operator= (const D &other)`
- `D(D &&other) noexcept`
- `D & operator= (D &&other) noexcept`
- `virtual ~D()`

### Открытые атрибуты

- `int IntegerD`
- `arma::mat MatrixD`

### Друзья

- `void swap(D< SizeX, SizeY > &lhs, D< SizeX, SizeY > &rhs) noexcept`

## 3.4.1 Конструктор(ы)

### 3.4.1.1 `D()` [1/3]

```
template<size_t SizeX, size_t SizeY>
D< SizeX, SizeY >::D ( ) [inline]
```

## 3.4.1.2 D() [2/3]

```
template<size_t SizeX, size_t SizeY>
D< SizeX, SizeY >::D (
    const D< SizeX, SizeY > & other )    [inline]
```

## 3.4.1.3 D() [3/3]

```
template<size_t SizeX, size_t SizeY>
D< SizeX, SizeY >::D (
    D< SizeX, SizeY > && other )    [inline], [noexcept]
```

## 3.4.1.4 ~D()

```
template<size_t SizeX, size_t SizeY>
virtual D< SizeX, SizeY >::~~D ( )    [inline], [virtual]
```

## 3.4.2 Методы

## 3.4.2.1 operator=() [1/2]

```
template<size_t SizeX, size_t SizeY>
D & D< SizeX, SizeY >::operator= (
    const D< SizeX, SizeY > & other )    [inline]
```

## 3.4.2.2 operator=() [2/2]

```
template<size_t SizeX, size_t SizeY>
D & D< SizeX, SizeY >::operator= (
    D< SizeX, SizeY > && other )    [inline], [noexcept]
```

## 3.4.3 Документация по друзьям класса и функциям, относящимся к классу

### 3.4.3.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
    D< SizeX, SizeY > & lhs,
    D< SizeX, SizeY > & rhs ) [friend]
```

## 3.4.4 Данные класса

### 3.4.4.1 IntegerD

```
template<size_t SizeX, size_t SizeY>
int D< SizeX, SizeY >::IntegerD
```

### 3.4.4.2 MatrixD

```
template<size_t SizeX, size_t SizeY>
arma::mat D< SizeX, SizeY >::MatrixD
```

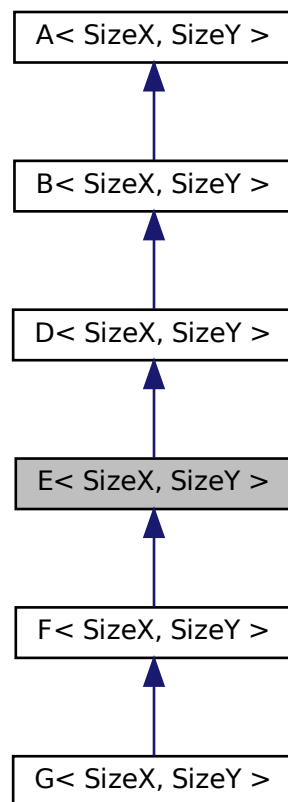
Объявления и описания членов класса находятся в файле:

- classes.h

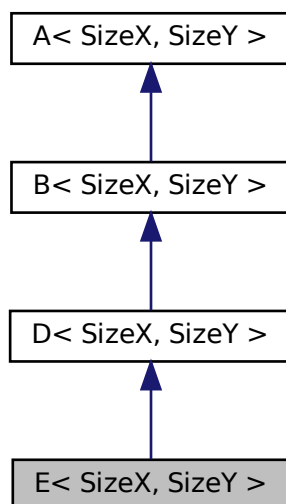
## 3.5 Шаблон класса E< SizeX, SizeY >

```
#include <classes.h>
```

Граф наследования: E< SizeX, SizeY >:



Граф связей класса `E< SizeX, SizeY >`:



#### Открытые члены

- `E()`
- `E(const E &other)`
- `E & operator= (const E &other)`
- `E(E &&other) noexcept`
- `E & operator= (E &&other) noexcept`
- `virtual ~E()`

#### Открытые атрибуты

- `int IntegerE`
- `arma::mat MatrixE`

#### Друзья

- `void swap(E< SizeX, SizeY > &lhs, E< SizeX, SizeY > &rhs) noexcept`

### 3.5.1 Конструктор(ы)

#### 3.5.1.1 `E()` [1/3]

```
template<size_t SizeX, size_t SizeY>
E< SizeX, SizeY >::E ( ) [inline]
```



## 3.5.1.2 E() [2/3]

```
template<size_t SizeX, size_t SizeY>
E< SizeX, SizeY >::E (
    const E< SizeX, SizeY > & other )    [inline]
```

## 3.5.1.3 E() [3/3]

```
template<size_t SizeX, size_t SizeY>
E< SizeX, SizeY >::E (
    E< SizeX, SizeY > && other )    [inline], [noexcept]
```

## 3.5.1.4 ~E()

```
template<size_t SizeX, size_t SizeY>
virtual E< SizeX, SizeY >::~~E ( )    [inline], [virtual]
```

## 3.5.2 Методы

## 3.5.2.1 operator=() [1/2]

```
template<size_t SizeX, size_t SizeY>
E & E< SizeX, SizeY >::operator= (
    const E< SizeX, SizeY > & other )    [inline]
```

## 3.5.2.2 operator=() [2/2]

```
template<size_t SizeX, size_t SizeY>
E & E< SizeX, SizeY >::operator= (
    E< SizeX, SizeY > && other )    [inline], [noexcept]
```

## 3.5.3 Документация по друзьям класса и функциям, относящимся к классу

### 3.5.3.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
    E< SizeX, SizeY > & lhs,
    E< SizeX, SizeY > & rhs ) [friend]
```

## 3.5.4 Данные класса

### 3.5.4.1 IntegerE

```
template<size_t SizeX, size_t SizeY>
int E< SizeX, SizeY >::IntegerE
```

### 3.5.4.2 MatrixE

```
template<size_t SizeX, size_t SizeY>
arma::mat E< SizeX, SizeY >::MatrixE
```

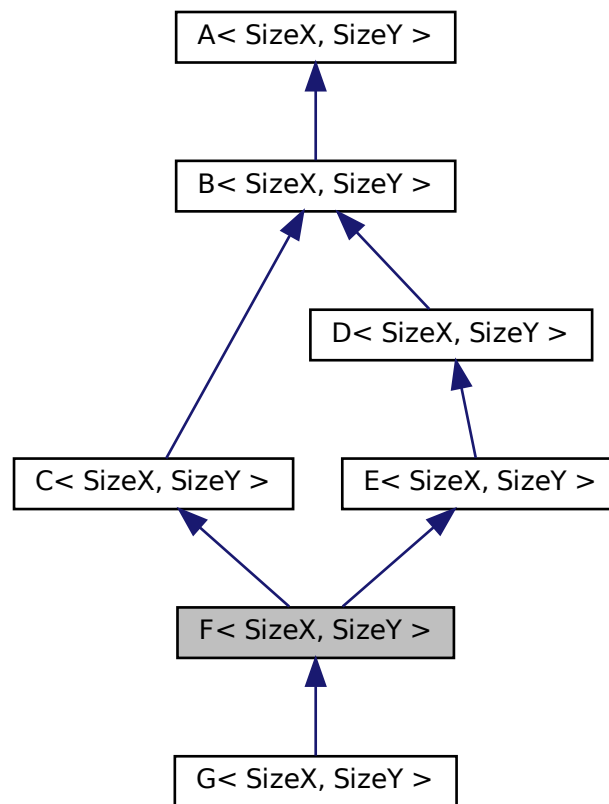
Объявления и описания членов класса находятся в файле:

- classes.h

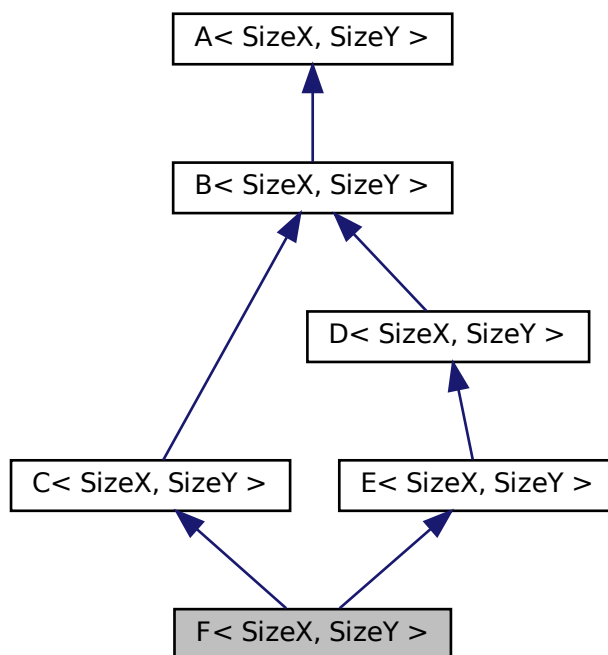
## 3.6 Шаблон класса F< SizeX, SizeY >

```
#include <classes.h>
```

Граф наследования:F< SizeX, SizeY >:



Граф связей класса `F< SizeX, SizeY >`:



#### Открытые члены

- `F ()`
- `F (const F &other)`
- `F & operator= (const F &other)`
- `F (F &&other) noexcept`
- `F & operator= (F &&other) noexcept`
- `virtual ~F ()`

#### Открытые атрибуты

- `int IntegerF`
- `arma::mat MatrixF`

#### Друзья

- `void swap (F< SizeX, SizeY > &lhs, F< SizeX, SizeY > &rhs) noexcept`

### 3.6.1 Конструктор(ы)

## 3.6.1.1 F() [1/3]

```
template<size_t SizeX, size_t SizeY>
F< SizeX, SizeY >::F ( ) [inline]
```

## 3.6.1.2 F() [2/3]

```
template<size_t SizeX, size_t SizeY>
F< SizeX, SizeY >::F (
    const F< SizeX, SizeY > & other ) [inline]
```

## 3.6.1.3 F() [3/3]

```
template<size_t SizeX, size_t SizeY>
F< SizeX, SizeY >::F (
    F< SizeX, SizeY > && other ) [inline], [noexcept]
```

## 3.6.1.4 ~F()

```
template<size_t SizeX, size_t SizeY>
virtual F< SizeX, SizeY >::~~F ( ) [inline], [virtual]
```

## 3.6.2 Методы

## 3.6.2.1 operator=() [1/2]

```
template<size_t SizeX, size_t SizeY>
F & F< SizeX, SizeY >::operator= (
    const F< SizeX, SizeY > & other ) [inline]
```

## 3.6.2.2 operator=() [2/2]

```
template<size_t SizeX, size_t SizeY>
F & F< SizeX, SizeY >::operator= (
    F< SizeX, SizeY > && other ) [inline], [noexcept]
```

### 3.6.3 Документация по друзьям класса и функциям, относящимся к классу

#### 3.6.3.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
    F< SizeX, SizeY > & lhs,
    F< SizeX, SizeY > & rhs ) [friend]
```

### 3.6.4 Данные класса

#### 3.6.4.1 IntegerF

```
template<size_t SizeX, size_t SizeY>
int F< SizeX, SizeY >::IntegerF
```

#### 3.6.4.2 MatrixF

```
template<size_t SizeX, size_t SizeY>
arma::mat F< SizeX, SizeY >::MatrixF
```

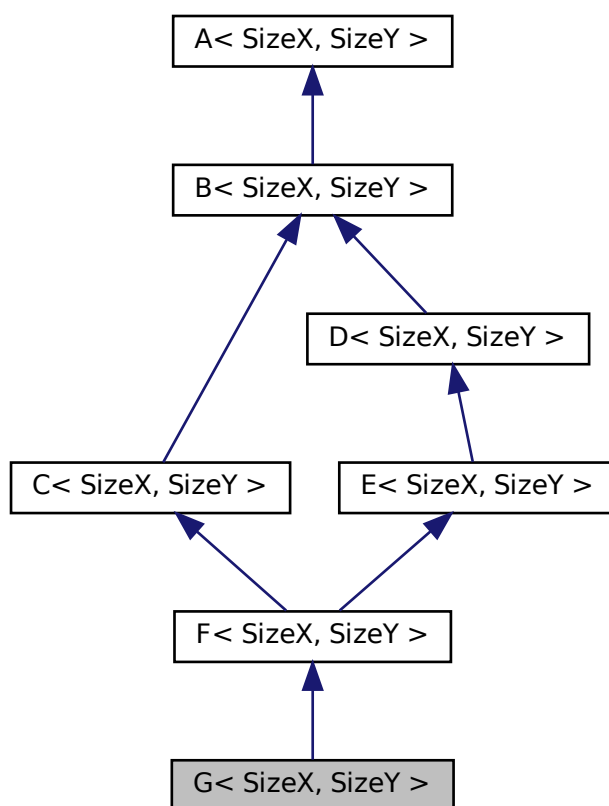
Объявления и описания членов класса находятся в файле:

- classes.h

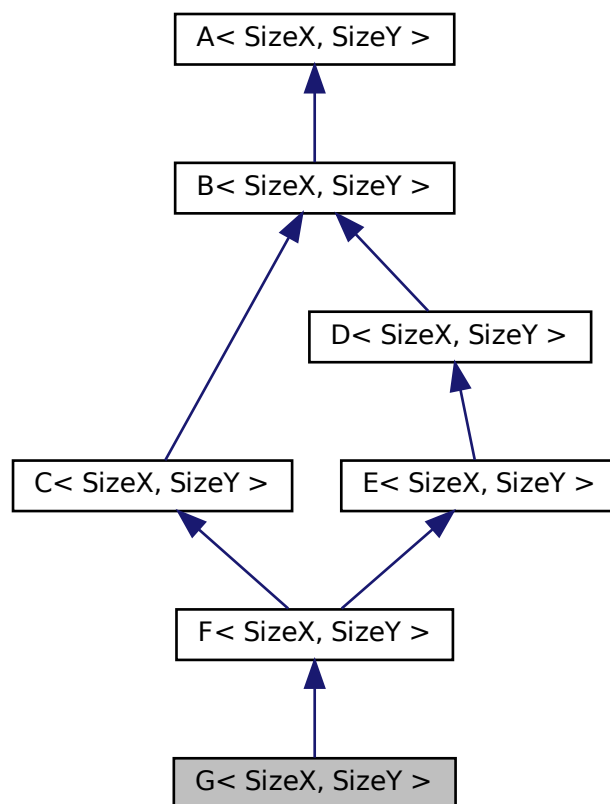
## 3.7 Шаблон класса G< SizeX, SizeY >

```
#include <classes.h>
```

Граф наследования: G< SizeX, SizeY >:



Граф связей класса `G< SizeX, SizeY >`:



### Открытые члены

- `G ()`
- `G (const G &other)`
- `G & operator= (const G &other)`
- `G (G &&other)`
- `G & operator= (G &&other)`
- `virtual ~G ()`

### Открытые атрибуты

- `int IntegerG`
- `arma::mat MatrixG`

### Друзья

- `void swap (G< SizeX, SizeY > &lhs, G< SizeX, SizeY > &rhs)`



### 3.7.1 Конструктор(ы)

#### 3.7.1.1 G() [1/3]

```
template<size_t SizeX, size_t SizeY>  
G< SizeX, SizeY >::G ( ) [inline]
```

#### 3.7.1.2 G() [2/3]

```
template<size_t SizeX, size_t SizeY>  
G< SizeX, SizeY >::G (   
    const G< SizeX, SizeY > & other ) [inline]
```

#### 3.7.1.3 G() [3/3]

```
template<size_t SizeX, size_t SizeY>  
G< SizeX, SizeY >::G (   
    G< SizeX, SizeY > && other ) [inline]
```

#### 3.7.1.4 ~G()

```
template<size_t SizeX, size_t SizeY>  
virtual G< SizeX, SizeY >::~G ( ) [inline], [virtual]
```

### 3.7.2 Методы

#### 3.7.2.1 operator=() [1/2]

```
template<size_t SizeX, size_t SizeY>  
G & G< SizeX, SizeY >::operator= (   
    const G< SizeX, SizeY > & other ) [inline]
```

### 3.7.2.2 operator=() [2/2]

```
template<size_t SizeX, size_t SizeY>
G & G< SizeX, SizeY >::operator= (
    G< SizeX, SizeY > && other ) [inline]
```

## 3.7.3 Документация по друзьям класса и функциям, относящимся к классу

### 3.7.3.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
    G< SizeX, SizeY > & lhs,
    G< SizeX, SizeY > & rhs ) [friend]
```

## 3.7.4 Данные класса

### 3.7.4.1 IntegerG

```
template<size_t SizeX, size_t SizeY>
int G< SizeX, SizeY >::IntegerG
```

### 3.7.4.2 MatrixG

```
template<size_t SizeX, size_t SizeY>
arma::mat G< SizeX, SizeY >::MatrixG
```

Объявления и описания членов класса находятся в файле:

- classes.h

# Предметный указатель

- ~A
  - A< SizeX, SizeY >, 6
- ~B
  - B< SizeX, SizeY >, 10
- ~C
  - C< SizeX, SizeY >, 13
- ~D
  - D< SizeX, SizeY >, 17
- ~E
  - E< SizeX, SizeY >, 21
- ~F
  - F< SizeX, SizeY >, 25
- ~G
  - G< SizeX, SizeY >, 29
- A
  - A< SizeX, SizeY >, 6
- A< SizeX, SizeY >, 5
  - ~A, 6
  - A, 6
  - IntegerA, 7
  - MatrixA, 7
  - operator=, 7
  - swap, 7
- B
  - B< SizeX, SizeY >, 9, 10
- B< SizeX, SizeY >, 8
  - ~B, 10
  - B, 9, 10
  - IntegerB, 11
  - MatrixB, 11
  - operator=, 10
  - swap, 10
- C
  - C< SizeX, SizeY >, 13
- C< SizeX, SizeY >, 11
  - ~C, 13
  - C, 13
  - IntegerC, 14
  - MatrixC, 14
  - operator=, 14
  - swap, 14
- D
  - D< SizeX, SizeY >, 16, 17
- D< SizeX, SizeY >, 15
  - ~D, 17
  - D, 16, 17
- IntegerD, 18
- MatrixD, 18
- operator=, 17
- swap, 17
- E
  - E< SizeX, SizeY >, 20, 21
- E< SizeX, SizeY >, 18
  - ~E, 21
  - E, 20, 21
  - IntegerE, 22
  - MatrixE, 22
  - operator=, 21
  - swap, 21
- F
  - F< SizeX, SizeY >, 24, 25
- F< SizeX, SizeY >, 22
  - ~F, 25
  - F, 24, 25
  - IntegerF, 26
  - MatrixF, 26
  - operator=, 25
  - swap, 26
- G
  - G< SizeX, SizeY >, 29
- G< SizeX, SizeY >, 26
  - ~G, 29
  - G, 29
  - IntegerG, 30
  - MatrixG, 30
  - operator=, 29
  - swap, 30
- IntegerA
  - A< SizeX, SizeY >, 7
- IntegerB
  - B< SizeX, SizeY >, 11
- IntegerC
  - C< SizeX, SizeY >, 14
- IntegerD
  - D< SizeX, SizeY >, 18
- IntegerE
  - E< SizeX, SizeY >, 22
- IntegerF
  - F< SizeX, SizeY >, 26
- IntegerG
  - G< SizeX, SizeY >, 30

## MatrixA

A< SizeX, SizeY >, [7](#)

## MatrixB

B< SizeX, SizeY >, [11](#)

## MatrixC

C< SizeX, SizeY >, [14](#)

## MatrixD

D< SizeX, SizeY >, [18](#)

## MatrixE

E< SizeX, SizeY >, [22](#)

## MatrixF

F< SizeX, SizeY >, [26](#)

## MatrixG

G< SizeX, SizeY >, [30](#)

## operator=

A< SizeX, SizeY >, [7](#)B< SizeX, SizeY >, [10](#)C< SizeX, SizeY >, [14](#)D< SizeX, SizeY >, [17](#)E< SizeX, SizeY >, [21](#)F< SizeX, SizeY >, [25](#)G< SizeX, SizeY >, [29](#)

## swap

A< SizeX, SizeY >, [7](#)B< SizeX, SizeY >, [10](#)C< SizeX, SizeY >, [14](#)D< SizeX, SizeY >, [17](#)E< SizeX, SizeY >, [21](#)F< SizeX, SizeY >, [26](#)G< SizeX, SizeY >, [30](#)