

My Project

Создано системой Doxygen 1.9.6



1 Алфавитный указатель классов	1
1.1 Классы	1
2 Список файлов	3
2.1 Файлы	3
3 Классы	5
3.1 Класс Book	5
3.1.1 Подробное описание	5
3.1.2 Конструктор(ы)	5
3.1.2.1 Book()	6
3.1.3 Методы	6
3.1.3.1 operator<()	6
3.1.3.2 operator<=()	6
3.1.3.3 operator==()	7
3.1.3.4 operator>()	7
3.1.3.5 operator>=()	7
3.1.4 Данные класса	8
3.1.4.1 authorName	8
3.1.4.2 bookName	8
3.1.4.3 pages	8
3.1.4.4 year	8
3.2 Шаблон класса Searching< T >	8
3.2.1 Подробное описание	9
3.2.2 Методы	9
3.2.2.1 binary()	9
3.2.2.2 linear()	9
3.3 Шаблон класса Sorting< T >	10
3.3.1 Подробное описание	10
3.3.2 Методы	10
3.3.2.1 quickSort()	10
4 Файлы	11
4.1 Файл D:/Lab2V2.cpp	11
4.1.1 Функции	11
4.1.1.1 main()	11
Предметный указатель	13



# Глава 1

## Алфавитный указатель классов

### 1.1 Классы

Классы с их кратким описанием.

<a href="#">Book</a>	Класс, задающий книгу в библиотеке . . . . .	5
<a href="#">Searching&lt; T &gt;</a>	Класс, содержащий алгоритмы поиска . . . . .	8
<a href="#">Sorting&lt; T &gt;</a>	Класс, содержащий реализуемые сортировки . . . . .	10



## Глава 2

# Список файлов

### 2.1 Файлы

Полный список файлов.

D:/	<a href="#">Lab2V2.cpp</a>	.....	11
-----	----------------------------	-------	----





## Глава 3

# Классы

### 3.1 Класс Book

Класс, задающий книгу в библиотеке

#### Открытые члены

- `Book` (`std::string` `authorName`="", `std::string` `bookName`="", `int` `year`=1970, `int` `pages`=300)
- `bool` `operator<` (`Book` &right)
- `bool` `operator>` (`Book` &right)
- `bool` `operator<=` (`Book` &right)
- `bool` `operator>=` (`Book` &right)
- `bool` `operator==` (`const Book` &right) `const`

#### Открытые атрибуты

- `std::string` `authorName`  
Имя автора
- `std::string` `bookName`  
Название книги
- `int` `year`  
Год публикации
- `int` `pages`  
Число страниц

#### 3.1.1 Подробное описание

Класс, задающий книгу в библиотеке

#### 3.1.2 Конструктор(ы)

### 3.1.2.1 Book()

```
Book::Book (
    std::string authorName = "",
    std::string bookName = "",
    int year = 1970,
    int pages = 300 ) [inline]
```

Создает объект [Book](#)

Аргументы

in	authorName	Имя автора
in	bookName	Название книги
in	year	Год публикации
in	pages	Число страниц

Возвращает

Возвращает объект [Book](#)

### 3.1.3 Методы

#### 3.1.3.1 operator<()

```
bool Book::operator< (
    Book & right ) [inline]
```

Оператор сравнения меньше

Аргументы

in	right	правое сравниваемое значение
----	-------	------------------------------

Возвращает

Возвращает true, если левый объект строго меньше, false - иначе

#### 3.1.3.2 operator<=()

```
bool Book::operator<= (
    Book & right ) [inline]
```

Оператор сравнения меньше или равно

Аргументы

in	right	правое сравниваемое значение
----	-------	------------------------------

Возвращает

Возвращает true, если левый объект меньше или равен, false - иначе

### 3.1.3.3 operator==()

```
bool Book::operator==(
    const Book & right ) const    [inline]
```

### 3.1.3.4 operator>()

```
bool Book::operator> (
    Book & right )    [inline]
```

Оператор сравнения больше

Аргументы

in	right	правое сравниваемое значение
----	-------	------------------------------

Возвращает

Возвращает true, если левый объект строго больше, false - иначе

### 3.1.3.5 operator>=()

```
bool Book::operator>= (
    Book & right )    [inline]
```

Оператор сравнения больше или равно

Аргументы

in	right	правое сравниваемое значение
----	-------	------------------------------

Возвращает

Возвращает true, если левый объект больше или равен, false - иначе

### 3.1.4 Данные класса

#### 3.1.4.1 authorName

`std::string Book::authorName`

Имя автора

#### 3.1.4.2 bookName

`std::string Book::bookName`

Название книги

#### 3.1.4.3 pages

`int Book::pages`

Число страниц

#### 3.1.4.4 year

`int Book::year`

Год публикации

Объявления и описания членов класса находятся в файле:

- [D:/Lab2V2.cpp](#)

## 3.2 Шаблон класса Searching< T >

Класс, содержащий алгоритмы поиска

## Открытые статические члены

- static std::vector< int > [linear](#) (T data[], int n, T &searching)
- static std::vector< int > [binary](#) (T data[], int n, T &searching)

### 3.2.1 Подробное описание

```
template<class T>
class Searching< T >
```

Класс, содержащий алгоритмы поиска

### 3.2.2 Методы

#### 3.2.2.1 binary()

```
template<class T >
static std::vector< int > Searching< T >::binary (
    T data[],
    int n,
    T & searching )  [inline], [static]
```

Бинарный поиск

Аргументы

in	data	Массив, в котором ищутся элементы
in	n	Число элементов в массиве
in	searching	Элемент, который мы ищем

#### 3.2.2.2 linear()

```
template<class T >
static std::vector< int > Searching< T >::linear (
    T data[],
    int n,
    T & searching )  [inline], [static]
```

Линейный поиск

Аргументы

in	data	Массив, в котором ищутся элементы
in	n	Число элементов в массиве
in	searching	Элемент, который мы ищем

Объявления и описания членов класса находятся в файле:

- D:/Lab2V2.cpp

### 3.3 Шаблон класса `Sorting< T >`

Класс, содержащий реализуемые сортировки

Открытые статические члены

- static void `quickSort` (T \*data, int n)

#### 3.3.1 Подробное описание

```
template<class T>
class Sorting< T >
```

Класс, содержащий реализуемые сортировки

#### 3.3.2 Методы

##### 3.3.2.1 `quickSort()`

```
template<class T >
static void Sorting< T >::quickSort (
    T * data,
    int n ) [inline], [static]
```

Оболочка для быстрой сортировки

Аргументы

in	data	Сортируемый массив
in	n	Число элементов в массиве

Объявления и описания членов класса находятся в файле:

- D:/Lab2V2.cpp

## Глава 4

# Файлы

### 4.1 Файл D:/Lab2V2.cpp

```
#include <iostream>
#include <fstream>
#include <string>
#include <vector>
#include <chrono>
```

#### Классы

- class [Book](#)  
Класс, задающий книгу в библиотеке
- class [Sorting< T >](#)  
Класс, содержащий реализуемые сортировки
- class [Searching< T >](#)  
Класс, содержащий алгоритмы поиска

#### Функции

- int [main](#) ()

#### 4.1.1 Функции

##### 4.1.1.1 main()

int main ( )

Основной метод программы

Возвращает

Код завершения программы





# Предметный указатель

authorName  
    Book, [8](#)

year  
    Book, [8](#)

binary  
    Searching< T >, [9](#)

Book, [5](#)  
    authorName, [8](#)  
    Book, [5](#)  
    bookName, [8](#)  
    operator<, [6](#)  
    operator<=, [6](#)  
    operator>, [7](#)  
    operator>=, [7](#)  
    operator==, [7](#)  
    pages, [8](#)  
    year, [8](#)

bookName  
    Book, [8](#)

D:/Lab2V2.cpp, [11](#)

Lab2V2.cpp  
    main, [11](#)

linear  
    Searching< T >, [9](#)

main  
    Lab2V2.cpp, [11](#)

operator<  
    Book, [6](#)

operator<=  
    Book, [6](#)

operator>  
    Book, [7](#)

operator>=  
    Book, [7](#)

operator==  
    Book, [7](#)

pages  
    Book, [8](#)

quickSort  
    Sorting< T >, [10](#)

Searching< T >, [8](#)  
    binary, [9](#)  
    linear, [9](#)

Sorting< T >, [10](#)  
    quickSort, [10](#)