

004.43
С83

ЯЗЫК C++

ПРОГРАММИРОВАНИЯ

специальное издание



Бьерн Страуструп
создатель C++



БИНОМ

ИЗДАНИЕ 2010

**The
C++
Programming
Language**

Special Edition

Bjarne Stroustrup

AT&T Labs
Murray Hill, New Jersey



Addison-Wesley

An Imprint of Addison Wesley Longman, Inc.

Бьери Страуструп

**Язык
программирования
C++**

Специальное издание

Перевод с английского
под редакцией
Н. Н. Мартынова



Москва
Издательство **БИНОМ**

2015

УДК 004.43
ББК 32.973.26-018.1
С83

Бьерн Страуструп

**Язык программирования C++. Специальное издание. Пер. с англ. — М.:
Издательство Бином, 2015 г. — 1136 с.: ил.**

Книга написана Бьерном Страуструпом - автором языка программирования C++ - и является каноническим изложением возможностей этого языка. Помимо подробного описания собственно языка, на страницах книги вы найдете доказавшие свою эффективность подходы к решению разнообразных задач проектирования и программирования. Многочисленные примеры демонстрируют как хороший стиль программирования на C-совместимом ядре C++, так и современный объектно-ориентированный подход к созданию программных продуктов.

Книга адресована программистам, использующим в своей повседневной работе C++. Она также будет полезна преподавателям, студентам и всем, кто хочет ознакомиться с описанием языка «из первых рук».

Все права защищены. Никакая часть этой книги не может быть воспроизведена в любой форме или любыми средствами, электронными или механическими, включая фотографирование, магнитную запись или иные средства копирования или сохранения информации без письменного разрешения издательства.

Translation copyright © 2010 by Binom Publishers.
C++ Programming Language, The: Special Edition,
First Edition by Bjarne Stroustrup, Copyright © 2000, All Rights Reserved.
Published by arrangement with the original publisher, Addison Wesley Longman, a Pearson Education Company.

ISBN 978-5-7989-0425-9 (рус.)
ISBN 0-201-70073-5 (англ.)

© Addison Wesley Longman,
a Pearson Education Company, 2000
© Издание на русском языке.
Издательство Бином, 2012

Краткое содержание

Предисловие переводчика и редактора.....	25
Предисловие автора к третьему русскому изданию.....	26
Предисловие.....	29
Предисловие ко второму изданию.....	31
Предисловие к первому изданию.....	33
Введение.....	35
1. Обращение к читателю.....	37
2. Обзор языка C++.....	59
3. Обзор стандартной библиотеки.....	85
Часть I. Основные средства.....	111
4. Типы и объявления.....	113
5. Указатели, массивы и структуры.....	133
6. Выражения и операторы.....	155
7. Функции.....	195
8. Пространства имен и исключения.....	219
9. Исходные файлы и программы.....	253
Часть II. Механизмы абстракции.....	281
10. Классы.....	283
11. Перегрузка операций.....	327
12. Наследование классов.....	371
13. Шаблоны.....	401
14. Обработка исключений.....	433
15. Иерархии классов.....	473
Часть III. Стандартная библиотека.....	515
16. Организация библиотеки и контейнеры.....	517
17. Стандартные контейнеры.....	555
18. Алгоритмы и классы функциональных объектов.....	607
19. Итераторы и аллокаторы.....	655
20. Строки.....	689
21. Потoki.....	717
22. Классы для математических вычислений.....	775

Часть IV. Проектирование с использованием C++.....	809
23. Общий взгляд на разработку программ. Проектирование.....	811
24. Проектирование и программирование.....	849
25. Роли классов.....	895
Приложения и предметный указатель.....	923
A. Г рамматика.....	925
B. Совместимость.....	947
C. Технические подробности.....	961
D. Локализация.....	1007
E. Исключения и безопасность стандартной библиотеки.....	1077
Предметный указатель.....	1117

Содержание

Предисловие переводчика и редактора.....	25
Предисловие автора к третьему русскому изданию.....	26
Предисловие.....	29
Предисловие ко второму изданию.....	31
Предисловие к первому изданию.....	33
Введение.....	35
Глава 1. Обращение к читателю.....	37
1.1. Структура книги.....	37
1.1.1. Примеры и ссылки.....	39
1.1.2. Упражнения.....	40
1.1.3. Замечания о конкретных реализациях языка (компиляторах).....	40
1.2. Как изучать С++.....	40
1.3. Как проектировался С++.....	42
1.3.1. Эффективность и структура.....	43
1.3.2. философские замечания.....	44
1.4. Исторические замечания.....	45
1.5. Применение С++.....	47
1.6. Языки С и С++.....	49
1.6.1. Информация для С-программистов.....	50
1.6.2. Информация для С++-программистов.....	50
1.7. Размышления о программировании на С++.....	51
1.8. Советы.....	53
1.8.1. Литература.....	54
Глава 2. Обзор языка С++.....	59
2.1. Что такое язык С++?.....	59
2.2. Парадигмы программирования.....	60
2.3. Процедурное программирование.....	61
2.3.1. Переменные и простейшая арифметика.....	62
2.3.2. Операторы ветвления и циклы.....	63
2.3.3. Указатели и массивы.....	64
2.4. Модульное программирование.....	65
2.4.1. Раздельная компиляция.....	67
2.4.2. Обработка исключений.....	68

2.5. Абстракция данных.....	69
2.5.1. Модули, определяющие типы.....	69
2.5.2. Типы, определяемые пользователем.....	71
2.5.3. Конкретные типы.....	73
2.5.4. Абстрактные типы.....	74
2.5.5. Виртуальные функции.....	77
2.6. Объектно-ориентированное программирование.....	77
2.6.1. Проблемы, связанные с конкретными типами.....	78
2.6.2. Классовые иерархии.....	79
2.7. Обобщенное программирование.....	81
2.7.1. Контейнеры.....	81
2.7.2. Обобщенные алгоритмы.....	82
2.8. Заключение.....	84
2.9. Советы.....	84
Глава 3. Обзор стандартной библиотеки.....	85
3.1. Введение.....	85
3.2. Hello, world! (Здравствуй, мир!).....	86
3.3. Пространствоимен стандартной библиотеки.....	87
3.4. Вывод.....	87
3.5. Строки.....	88
3.5.1. C-строки.....	90
3.6. Ввод.....	90
3.7. Контейнеры.....	92
3.7.1. Контейнер vector.....	93
3.7.2. Проверка диапазона индексов.....	94
3.7.3. Контейнер list.....	95
3.7.4. Контейнер map.....	96
3.7.5. Контейнеры стандартной библиотеки.....	97
3.8. Алгоритмы.....	98
3.8.1. Использование итераторов.....	99
3.8.2. Типы итераторов.....	101
3.8.3. Итераторы и ввод/вывод.....	102
3.8.4. Алгоритм for __ each и предикаты.....	103
3.8.5. Алгоритмы, использующие функции-члены классов.....	105
3.8.6. Алгоритмы стандартной библиотеки.....	106
3.9. Математические вычисления.....	107
3.9.1. Комплексные числа.....	107
3.9.2. Векторная арифметика.....	107
3.9.3. Поддержка базовых вычислительных операций.....	108
3.10. Основные средства стандартной библиотеки.....	108
3.11. Советы.....	109
Часть I. Основные средства.....	111
Глава 4. Типы и объявления.....	113
4.1. Типы.....	113
4.1.1. Фундаментальные типы.....	114
4.2. Логический тип.....	115

4.3. Символьные типы.....	116
4.3.1. Символьные литералы.....	117
4.4. Целые типы.....	118
4.4.1. Целые литералы.....	118
4.5. Типы с плавающей запятой.....	119
4.5.1. Литералы с плавающей запятой.....	119
4.6. Размеры.....	119
4.7. Тип void.....	121
4.8. Перечисления.....	122
4.9. Объявления.....	123
4.9.1. Структура объявления.....	125
4.9.2. Объявление нескольких имен.....	126
4.9.3. Имена.....	126
4.9.4. Область видимости.....	127
4.9.5. Инициализация.....	129
4.9.6. Объекты и леводопустимые выражения (lvalue).....	130
4.9.7. Ключевое слово typedef.....	130
4.10. Советы.....	131
4.11. Упражнения.....	132
Глава 5. Указатели, массивы и структуры.....	133
5.1. Указатели.....	133
5.1.1. Ноль.....	134
5.2. Массивы.....	135
5.2.1. Инициализация массивов.....	135
5.2.2. Строковые литералы.....	136
5.3. Указатели на массивы.....	138
5.3.1. Доступ к элементам массивов.....	139
5.4. Константы.....	141
5.4.1. Указатели и константы.....	143
5.5. Ссылки.....	144
5.6. Тип void*.....	148
5.7. Структуры.....	149
5.7.1. Эквивалентность типов.....	152
5.8. Советы.....	152
5.9. Упражнения.....	152
Глава 6. Выражения и операторы.....	155
6.1. Калькулятор.....	155
6.1.1. Синтаксический анализатор.....	156
6.1.2. Функция ввода.....	161
6.1.3. Низкоуровневый ввод.....	163
6.1.4. Обработка ошибок.....	164
6.1.5. Управляющая программа.....	165
6.1.6. Заголовочные файлы.....	166
6.1.7. Аргументы командной строки.....	167
6.1.8. Замечания о стиле.....	169
6.2. Обзор операций языка C++.....	169
6.2.1. Результаты операций.....	171

6.2.2. Последовательность вычислений.....	172
6.2.3. Приоритет операций.....	173
6.2.4. Побитовые логические операции.....	174
6.2.5. Инкремент и декремент.....	175
6.2.6. Свободная память.....	177
6.2.6.1. Массивы.....	179
6.2.6.2. Исчерпание памяти.....	180
6.2.7. Явное приведение типов.....	181
6.2.8. Конструкторы.....	182
6.3. Обзор операторов языка C++.....	183
6.3.1. Объявления как операторы.....	184
6.3.2. Операторы выбора (условные операторы).....	185
6.3.2.1. Объявления в условиях.....	187
6.3.3. Операторы цикла.....	188
6.3.3.1. Объявления в операторах цикла for.....	189
6.3.4. Оператор goto.....	189
6.4. Комментарии и отступы.....	190
6.5. Советы.....	192
6.6. Упражнения.....	192
Глава 7. Функции.....	195
7.1. Объявления функций.....	195
7.1.1. Определения функций.....	196
7.1.2. Статические переменные.....	197
7.2. Передача аргументов.....	197
7.2.1. Массивы в качестве аргументов.....	199
7.3. Возвращаемое значение.....	200
7.4. Перегрузка имен функций.....	201
7.4.1. Перегрузка и возвращаемые типы.....	204
7.4.2. Перегрузка и области видимости.....	204
7.4.3. Явное разрешение неоднозначностей.....	204
7.4.4. Разрешение в случае нескольких аргументов.....	205
7.5. Аргументы по умолчанию.....	206
7.6. Неуказанное число аргументов.....	207
7.7. Указатели на функции.....	209
7.8. Макросы.....	213
7.8.1. Условная компиляция.....	215
7.9. Советы.....	216
7.10. Упражнения.....	217
Глава 8. Пространства имен и исключения.....	219
8.1. Разбиение на модули и интерфейсы.....	219
8.2. Пространства имен.....	221
8.2.1. Квалифицированные имена.....	223
8.2.2. Объявления using.....	224
8.2.3. Директивы using.....	226
8.2.4. Множественные интерфейсы.....	227
8.2.4.1. Альтернативы интерфейсам.....	229
8.2.5. Как избежать конфликта имен.....	231
8.2.5.1. Неименованные пространства имен.....	232

8.2.6. Поиск имен.....	233
8.2.7. Псевдонимы пространств имен.....	234
8.2.8. Композиция пространств имен.....	235
8.2.8.1. Отбор отдельных элементов из пространства имен.....	236
8.2.8.2. Композиция пространств имен и отбор элементов.....	237
8.2.9. Пространства имен и старый код.....	238
8.2.9.1. Пространства имен и язык C.....	238
8.2.9.2. Пространства имен и перегрузка.....	239
8.2.9.3. Пространства имен открыты.....	240
8.3. Исключения.....	241
8.3.1. Ключевые слова throw и catch.....	242
8.3.2. Обработка нескольких исключений.....	244
8.3.3. Исключения в программе калькулятора.....	246
8.3.3.1. Альтернативные стратегии обработки ошибок.....	249
8.4. Советы.....	251
8.5. Упражнения.....	252
Глава 9. Исходные файлы и программы.....	253
9.1. Раздельная компиляция.....	253
9.2. Компоновка (linkage).....	254
9.2.1. Заголовочные файлы.....	257
9.2.2. Заголовочные файлы стандартной библиотеки.....	259
9.2.3. Правило одного определения.....	260
9.2.4. Компоновка с кодом, написанном не на языке C++.....	262
9.2.5. Компоновка и указатели на функции.....	264
9.3. Применяем заголовочные файлы.....	265
9.3.1. Единственный заголовочный файл.....	265
9.3.2. Несколько заголовочных файлов.....	268
9.3.2.1. Остальные модули калькулятора.....	271
9.3.2.2. Использование заголовочных файлов.....	273
9.3.3. Защита от повторных включений.....	274
9.4. Программы.....	275
9.4.1. Инициализация нелокальных переменных.....	275
9.4.1.1. Завершение выполнения программы.....	276
9.5. Советы.....	278
9.6. Упражнения.....	278
Часть II. Механизмы абстракции.....	281
Глава 10. Классы.....	283
10.1. Введение.....	283
10.2. Классы.....	284
10.2.1. Функции-члены.....	284
10.2.2. Управление режимом доступа.....	285
10.2.3. Конструкторы.....	287
10.2.4. Статические члены.....	288
10.2.5. Копирование объектов класса.....	290
10.2.6. Константные функции-члены.....	290
10.2.7. Ссылки на себя.....	291
10.2.7.1. Физическое и логическое постоянство.....	292
10.2.7.2. Ключевое слово mutable.....	294

10.2.8. Структуры и классы.....	295
10.2.9. Определение функций в теле определения класса.....	296
10.3. Эффективные пользовательские типы.....	297
10.3.1. Функции-члены.....	300
10.3.2. Функции поддержки (helper functions).....	302
10.3.3. Перегруженные операции.....	303
10.3.4. Роль конкретных классов.....	303
10.4. Объекты.....	304
10.4.1. Деструкторы.....	305
10.4.2. Конструкторы по умолчанию.....	306
10.4.3. Конструирование и уничтожение объектов.....	307
10.4.4. Локальные объекты.....	307
10.4.4.1. Копирование объектов.....	308
10.4.5. Динамическое создание объектов в свободной памяти.....	309
10.4.6. Классовые объекты как члены классов.....	310
10.4.6.1. Обязательная инициализация членов.....	311
10.4.6.2. Члены-константы.....	312
10.4.6.3. Копирование членов.....	313
10.4.7. Массивы.....	314
10.4.8. Локальные статические объекты.....	315
10.4.9. Нелокальные объекты.....	316
10.4.10. Временные объекты.....	318
10.4.11. Размещение объектов в заданных блоках памяти.....	319
10.4.12. Объединения.....	321
10.5. Советы.....	322
10.6. Упражнения.....	323
Глава 11. Перегрузка операций.....	327
11.1. Введение.....	327
11.2. Функции-операции.....	329
11.2.1. Бинарные и унарные операции.....	330
11.2.2. Предопределенный смысл операций.....	331
11.2.3. Операции и пользовательские типы.....	331
11.2.4. Операции и пространства имен.....	332
11.3. Тип комплексных чисел.....	334
11.3.1. Перегрузка операций функциями-членами и глобальными функциями.....	334
11.3.2. Смешанная арифметика.....	336
11.3.3. Инициализация.....	337
11.3.4. Копирование.....	338
11.3.5. Конструкторы и преобразования типов.....	339
11.3.6. Литералы.....	340
11.3.7. Дополнительные функции-члены.....	341
11.3.8. Функции поддержки (helper functions).....	341
11.4. Операции приведения типов.....	342
11.4.1. Неоднозначности.....	344
11.5. Друзья класса.....	346
11.5.1. Поиск друзей.....	348
11.5.2. Функции-члены или друзья?.....	349
11.6. Объекты больших размеров.....	350

11.7. Важные операции.....	352
11.7.1. Конструктор с модификатором explicit.....	353
11.8. Индексирование.....	355
11.9. Функциональный вызов.....	356
11.10. Разыменование.....	358
11.11. Инкремент и декремент.....	360
11.12. Класс строк.....	362
11.13. Советы.....	367
11.14. Упражнения.....	368
Глава 12. Наследование классов.....	371
12.1. Введение.....	371
12.2. Производные классы.....	372
12.2.1. Функции-члены.....	375
12.2.2. Конструкторы и деструкторы.....	376
12.2.3. Копирование.....	378
12.2.4. Иерархии классов.....	378
12.2.5. Поля типа.....	379
12.2.6. Виртуальные функции.....	381
12.3. Абстрактные классы.....	384
12.4. Проектирование иерархий классов.....	386
12.4.1. Традиционные иерархии классов.....	387
12.4.1.1. Критика.....	389
12.4.2. Абстрактные классы.....	390
12.4.3. Альтернативные реализации.....	393
12.4.3.1. Критика.....	395
12.4.4. Локализация создания объектов.....	395
12.5. Классовые иерархии и абстрактные классы.....	397
12.6. Советы.....	397
12.7. Упражнения.....	398
Глава 13. Шаблоны.....	401
13.1. Введение.....	401
13.2. Простой шаблон строк.....	402
13.2.1. Определение шаблона.....	404
13.2.2. Конкретизация шаблона (template instantiation).....	406
13.2.3. Параметры шаблонов.....	406
13.2.4. Эквивалентность типов.....	407
13.2.5. Проверка типов.....	408
13.3. Шаблоны функций.....	409
13.3.1. Аргументы функциональных шаблонов.....	410
13.3.2. Перегрузка функциональных шаблонов.....	411
13.4. Применение аргументов шаблона для формирования различных вариантов поведения кода.....	414
13.4.1. Параметры шаблонов по умолчанию.....	415
13.5. Специализация.....	417
13.5.1. Порядок специализаций.....	420
13.5.2. Специализация шаблонов функций.....	420
13.6. Наследование и шаблоны.....	422

13.6.1. Параметризация и наследование.....	424
13.6.2. Шаблонные члены шаблонов.....	424
13.6.3. Отношения наследования.....	425
13.6.3.1. Преобразования шаблонов.....	426
13.7. Организация исходного кода.....	427
13.8. Советы.....	430
13.9. Упражнения.....	431
Глава 14. Обработка исключений.....	433
14.1. Обработка ошибок.....	433
14.1.1. Альтернативный взгляд на исключения».....	436
14.2. Группировка исключений.....	437
14.2.1. Производные исключения.....	438
14.2.2. Композитные (комбинированные) исключения.....	440
14.3. Перехват исключений.....	441
14.3.1. Повторная генерация исключений.....	441
14.3.2. Перехват любых исключений.....	442
14.3.2.1. Порядок записи обработчиков.....	443
14.4. Управление ресурсами.....	444
14.4.1. Использование конструкторов и деструкторов.....	446
14.4.2. Auto_ptr.....	447
14.4.3. Предостережение.....	449
14.4.4. Исключения и операция new.....	449
14.4.5. Исчерпание ресурсов.....	450
14.4.6. Исключения в конструкторах.....	452
14.4.6.1. Исключения и инициализация членов классов.....	454
14.4.6.2. Исключения и копирование.....	454
14.4.7. Исключения в деструкторах.....	455
14.5. Исключения, не являющиеся ошибками.....	455
14.6. Спецификация исключений.....	457
14.6.1. Проверка спецификации исключений.....	458
14.6.2. Неожиданные исключения.....	459
14.6.3. Отображение исключений.....	460
14.6.3.1. Отображение исключений пользователем.....	460
14.6.3.2. Восстановление типа исключения.....	461
14.7. Неперехваченные исключения.....	462
14.8. Исключения и эффективность.....	464
14.9. Альтернативы обработке ошибок.....	465
14.10. Стандартные исключения.....	467
14.11. Советы.....	469
14.12. Упражнения.....	470
Глава 15. Иерархии классов.....	473
15.1. Введение и обзор.....	473
15.2. Множественное наследование.....	474
15.2.1. Разрешение неоднозначности.....	475
15.2.2. Наследование и using-объявление.....	477
15.2.3. Повторяющиеся базовые классы.....	478
15.2.3.1. Замещение.....	479
15.2.4. Виртуальные базовые классы.....	480

15.2.4.1. Программирование виртуальных базовых классов.....	482
15.2.5. Применение множественного наследования.....	484
15.2.5.1. Замещение функций виртуальных базовых классов.....	486
15.3. Контроль доступа.....	487
15.3.1. Защищенные члены классов.....	489
15.3.1.1. Применение защищенных членов класса.....	490
15.3.2. Доступ к базовым классам.....	491
15.3.2.1. Множественное наследование и контроль доступа.....	492
15.3.2.2. Множественное наследование и контроль доступа.....	493
15.4. Механизм RTTI (Run-Time Type Information).....	493
15.4.1. Операция <code>dynamic_cast</code>	495
15.4.1.1. Применение <code>dynamic_cast</code> к ссылкам.....	497
15.4.2. Навигация по иерархиям классов.....	498
15.4.2.1. Операции <code>static_cast</code> и <code>dynamic_cast</code>	499
15.4.3. Конструирование и уничтожение классовых объектов.....	501
15.4.4. Операция <code>typeid</code> и расширенная информация о типе.....	501
15.4.4.1. Расширенная информация о типе.....	502
15.4.5. Корректное и некорректное применение RTTI.....	504
15.5. Указатели на члены классов.....	505
15.5.1. Базовые и производные классы.....	508
15.6. Свободная память.....	509
15.6.1. Выделение памяти под массивы.....	511
15.6.2. «Виртуальные конструкторы».....	511
15.7. Советы.....	513
15.8. Упражнения.....	514
Часть III. Стандартная библиотека.....	515
Глава 16. Организация библиотеки и контейнеры.....	517
16.1. Проектные решения стандартной библиотеки.....	517
16.1.1. Проектные ограничения.....	518
16.1.2. Организация стандартной библиотеки.....	520
16.1.3. Непосредственная поддержка языка C++.....	523
16.2. Дизайн контейнеров.....	524
16.2.1. Специализированные контейнеры и итераторы.....	524
16.2.2. Контейнеры с общим базовым классом.....	527
16.2.3. Контейнеры STL.....	531
16.3. Контейнер типа <code>vector</code>	533
16.3.1. Типы.....	533
16.3.2. Итераторы.....	535
16.3.3. Доступ к элементам.....	536
16.3.4. Конструкторы.....	538
16.3.5. Стековые операции.....	541
16.3.6. Операции над векторами, характерные для списков.....	543
16.3.7. Адресация элементов.....	546
16.3.8. Размер и емкость.....	547
16.3.9. Другие функции-члены.....	549
16.3.10. Вспомогательные функции (helper functions).....	550
16.3.11. Специализация <code>vector<bool></code>	550
16.4. Советы.....	551
16.5. Упражнения.....	552

Глава 17. Стандартные контейнеры	555
17.1. Стандартные контейнеры.....	555
17.1.1. Обзор контейнерных операций.....	556
17.1.2. Краткий обзор контейнеров.....	559
17.1.3. Внутреннее представление.....	560
17.1.4. Требования к элементам контейнеров.....	561
17.1.4.1. Операция сравнения "<".....	562
17.1.4.2. Другие операции сравнения.....	564
17.2. Последовательные контейнеры.....	565
17.2.1. Контейнер vector.....	565
17.2.2. Контейнер list.....	565
17.2.2.1. Операции splice(), sort() и merge().....	566
17.2.2.2. «Головные» операции.....	568
17.2.2.3. Другие операции.....	568
17.2.3. Контейнер deque.....	570
17.3. Адаптеры последовательных контейнеров.....	570
17.3.1. Стек.....	571
17.3.2. Очередь.....	572
17.3.3. Очередь с приоритетом.....	574
17.4. Ассоциативные контейнеры.....	576
17.4.1. Ассоциативный массив шар.....	576
17.4.1.1. Типы.....	577
17.4.1.2. Итераторы.....	577
17.4.1.3. Индексация.....	579
17.4.1.4. Конструкторы.....	581
17.4.1.5. Сравнения.....	581
17.4.1.6. Специфические для контейнера map операции.....	582
17.4.1.7. Операции, характерные для списков.....	584
17.4.1.8. Другие функции.....	586
17.4.2. Ассоциативный контейнер multimap.....	587
17.4.3. Ассоциативный контейнер set.....	588
17.4.4. Ассоциативный контейнер multiset.....	588
17.5. «Почти контейнеры».....	589
17.5.1. Строки типа string.....	589
17.5.2. Массивы valarray.....	589
17.5.3. Битовые поля bitset.....	589
17.5.3.1. Конструкторы.....	590
17.5.3.2. Побитовые операции.....	591
17.5.3.3. Прочие операции.....	592
17.5.4. Встроенные массивы.....	593
17.6. Создание нового контейнера.....	594
17.6.1. Контейнер hash_map.....	594
17.6.2. Представление и конструирование.....	596
17.6.2.1. Поиск.....	598
17.6.2.2. Операции erase() и resize().....	599
17.6.2.3. Хэширование.....	600
17.6.3. Другие хэшированные ассоциативные контейнеры.....	601
17.7. Советы.....	602
17.8. Упражнения.....	603

Глава 18. Алгоритмы и классы функциональных объектов	607
18.1. Введение.....	607
18.2. Обзор алгоритмов стандартной библиотеки.....	608
18.3. Диапазоны (интервалы) и контейнеры.....	613
18.3.1. Входные диапазоны.....	614
18.4. Классы функциональных объектов.....	615
18.4.1. Базовые классы функциональных объектов.....	616
18.4.2. Предикаты.....	617
18.4.2.1. Обзор предикатов.....	618
18.4.3. «Арифметические» функциональные объекты.....	619
18.4.4. Адаптеры («связывающие», для адаптации функций-членов и указателей на функции, «отрицающие»).....	620
18.4.4.1. «Связывающие» адаптеры.....	621
18.4.4.2. Адаптация функций-членов.....	623
18.4.4.3. Версии адаптеров для указателей на функции.....	624
18.4.4.4. «Отрицатели».....	625
18.5. Немодифицирующие алгоритмы.....	626
18.5.1. Алгоритм <code>for_each()</code>	626
18.5.2. Алгоритмы поиска.....	627
18.5.3. Алгоритмы <code>count()</code> и <code>count_if()</code>	629
18.5.4. Алгоритмы <code>equal()</code> и <code>mismatch()</code>	630
18.5.5. Поисковые алгоритмы.....	631
18.6. Модифицирующие алгоритмы.....	632
18.6.1. Копирующие алгоритмы.....	632
18.6.2. Алгоритм <code>transform()</code>	634
18.6.3. Алгоритм <code>unique()</code>	636
18.6.3.1. Критерии сортировки.....	637
18.6.4. Алгоритмы замены элементов.....	638
18.6.5. Алгоритмы удаления элементов.....	640
18.6.6. Алгоритмы <code>fill()</code> и <code>generate()</code>	640
18.6.7. Алгоритмы <code>reverse()</code> и <code>rotate()</code>	641
18.6.8. Обмен элементов последовательностей местами.....	642
18.7. Сортировка последовательностей.....	643
18.7.1. Сортировка.....	643
18.7.2. Бинарный поиск.....	644
18.7.3. Слияние (алгоритмы семейства <code>merge</code>).....	645
18.7.4. Разбиение элементов на две группы (алгоритмы семейства <code>partition</code>).....	646
18.7.5. Операции над множествами.....	646
18.8. «Кучи».....	648
18.9. Нахождение минимума и максимума.....	648
18.10. Перестановки (<code>permutations</code>).....	650
18.11. Алгоритмы в С-стиле.....	650
18.12. Советы.....	651
18.13. Упражнения.....	652
Глава 19. Итераторы и аллокаторы	655
19.1. Введение.....	655
19.2. Итераторы и последовательности.....	656
19.2.1. Операции над итераторами.....	656

19.2.2. Шаблон <code>iterator_traits</code>	658
19.2.3. Категории итераторов.....	660
19.2.4. Итераторы для вставок.....	662
19.2.5. Обратные итераторы.....	663
19.2.6. Поточные итераторы.....	664
19.2.6.1. Буфера потоков.....	666
19.3. Итераторы с проверкой.....	668
19.3.1. Исключения, контейнеры и алгоритмы.....	673
19.4. Аллокаторы (распределители памяти).....	674
19.4.1. Стандартный аллокатор.....	674
19.4.2. Пользовательский аллокатор.....	678
19.4.3. Обобщенные аллокаторы.....	680
19.4.4. Неинициализированная память.....	682
19.4.5. Динамическая память.....	684
19.4.6. Выделение памяти в стиле языка C.....	685
19.5. Советы.....	686
19.6. Упражнения.....	687
Глава 20. Строки	689
20.1. Введение.....	689
20.2. Символы.....	690
20.2.1. Шаблон <code>char_traits</code>	690
20.3. Стандартный строковый шаблон <code>basic_string</code>	692
20.3.1. Типы.....	693
20.3.2. Итераторы.....	694
20.3.3. Доступ к элементам (символам).....	695
20.3.4. Конструкторы.....	695
20.3.5. Ошибки.....	697
20.3.6. Присваивание.....	698
20.3.7. Преобразование в C-строку.....	699
20.3.8. Сравнения.....	701
20.3.9. Вставка.....	703
20.3.10. Конкатенация.....	704
20.3.11. Поиск.....	705
20.3.12. Замена.....	706
20.3.13. Подстроки.....	707
20.3.14. Размер и емкость.....	708
20.3.15. Операции ввода/вывода.....	709
20.3.16. Обмен строк.....	709
20.4. Стандартная библиотека языка C.....	710
20.4.1. C-строки.....	710
20.4.2. Классификация символов.....	712
20.5. Советы.....	713
20.6. Упражнения.....	714
Глава 21. Потоки	717
21.1. Введение.....	717
21.2. Вывод.....	719
21.2.1. Потоки вывода.....	720

21.2.2. Вывод встроенных типов.....	722
21.2.3. Вывод пользовательских типов.....	724
21.2.3.1. Виртуальные функции вывода.....	725
21.3. Ввод.....	726
21.3.1. Потоки ввода.....	726
21.3.2. Ввод встроенных типов.....	727
21.3.3. Состояние потока.....	729
21.3.4. Ввод символов.....	731
21.3.5. Ввод пользовательских типов.....	734
21.3.6. Исключения.....	735
21.3.7. Связывание потоков.....	737
21.3.8. Часовые (sentries).....	738
21.4. Форматирование.....	739
21.4.1. Состояние форматирования.....	739
21.4.1.1. Копирование состояния форматирования.....	741
21.4.2. Вывод целых.....	741
21.4.3. Вывод значений с плавающей запятой.....	742
21.4.4. Поля вывода.....	743
21.4.5. Выравнивание полей.....	744
21.4.6. Манипуляторы.....	745
21.4.6.1. Манипуляторы, принимающие аргументы.....	746
21.4.6.2. Стандартные манипуляторы ввода/вывода.....	747
21.4.6.3. Манипуляторы, определяемые пользователем.....	749
21.5. Файловые и строковые потоки.....	751
21.5.1. Файловые потоки.....	752
21.5.2. Закрытие потоков.....	753
21.5.3. Строковые потоки.....	755
21.6. Буферирование.....	756
21.6.1. Потоки вывода и буферы.....	757
21.6.2. Потоки ввода и буферы.....	758
21.6.3. Потоки и буферы.....	759
21.6.4. Буфера потоков.....	760
21.7. Локализация (интернационализация).....	764
21.7.1. Функции обратного вызова для потоков.....	766
21.8. Ввод/вывод языка C.....	766
21.9. Советы.....	770
21.10. Упражнения.....	771
Глава 22. Классы для математических вычислений.....	775
22.1. Введение.....	775
22.2. Числовые пределы.....	776
22.2.1. Макросы для предельных значений.....	778
22.3. Стандартные математические функции.....	778
22.4. Векторная арифметика.....	780
22.4.1. Конструкторы класса <code>valarray</code>	780
22.4.2. Индексирование и присваивание в классе <code>valarray</code>	782
22.4.3. Функции-члены.....	783
22.4.4. Внешние операции и функции.....	786
22.4.5. Срезы.....	786

22.4.6. Массив slice_array.....	789
22.4.7. Временные объекты, копирование, циклы.....	793
22.4.8. Обобщенные срезы.....	796
22.4.9. Маски (тип mask_array).....	797
22.4.10. Тип indirect_array.....	798
22.5. Комплексная арифметика.....	798
22.6. Обобщенные численные алгоритмы.....	800
22.6.1. Алгоритм accumulate().....	801
22.6.2. Алгоритм inner_product().....	802
22.6.3. Приращения (incremental changes).....	803
22.7. Случайные числа.....	804
22.8. Советы.....	806
22.9. Упражнения.....	807
Часть IV. Проектирование с использованием C++.....	809
Глава 23. Общий взгляд на разработку программ. Проектирование.....	811
23.1. Обзор.....	811
23.2. Введение.....	812
23.3. Цели и средства.....	815
23.4. Процесс разработки.....	818
23.4.1. Цикл разработки.....	820
23.4.2. Цели проектирования.....	822
23.4.3. Этапы проектирования.....	824
23.4.3.1. Этап 1: отражение концепций классами.....	825
23.4.3.2. Этап 2: определение операций.....	828
23.4.3.3. Этап 3: выявление зависимостей.....	830
23.4.3.4. Этап 4: определение интерфейсов.....	830
23.4.3.5. Реорганизация иерархии классов.....	831
23.4.3.6. Использование модельных образцов.....	832
23.4.4. Экспериментирование и анализ.....	834
23.4.5. Тестирование.....	836
23.4.6. Сопровождение и поддержка программ.....	837
23.4.7. Эффективность.....	837
23.5. Отдельные аспекты управления проектами.....	838
23.5.1. Повторное использование кода.....	838
23.5.2. Масштаб.....	840
23.5.3. Личности.....	841
23.5.4. Гибридное проектирование.....	843
23.6. Аннотированная библиография.....	845
23.7. Советы.....	847
Глава 24. Проектирование и программирование.....	849
24.1. Обзор.....	849
24.2. Проектирование и язык программирования.....	850
24.2.1. Отказ от классов.....	852
24.2.2. Отказ от производных классов и виртуальных функций.....	854
24.2.3. Игнорирование возможностей статической проверки типов.....	854
24.2.4. Отказ от традиционного программирования.....	857
24.2.5. Применение исключительно классовых иерархий наследования.....	859

24.3. Классы.....	860
24.3.1. Что представляют собой классы.....	861
24.3.2. Иерархии классов.....	862
24.3.2.1. Зависимости внутри иерархии классов.....	865
24.3.3. Агрегация (отношение включения).....	867
24.3.4. Агрегация и наследование.....	869
24.3.4.1. Альтернатива «включение/наследование».....	871
24.3.4.2. Альтернатива «агрегация/наследование».....	873
24.3.5. Отношение использования.....	874
24.3.6. Программируемые отношения.....	875
24.3.7. Отношения внутри класса.....	877
24.3.7.1. Инварианты.....	877
24.3.7.2. Утверждения.....	379
24.3.7.3. Предусловия и постусловия.....	882
24.3.7.4. Инкапсуляция.....	883
24.4. Компоненты.....	884
24.4.1. Шаблоны.....	886
24.4.2. Интерфейсы и реализации.....	888
24.4.3. «Жирные» интерфейсы.....	890
24.5. Советы.....	893
Глава 25. Роли классов.....	895
25.1. Разновидности классов.....	895
25.2. Конкретные классы (типы).....	896
25.2.1. Многократное использование конкретных типов.....	899
25.3. Абстрактные типы.....	900
25.4. Узловые классы.....	903
25.4.1. Изменение интерфейсов.....	905
25.5. Операции.....	908
25.6. Интерфейсные классы.....	909
25.6.1. Подгонка интерфейсов.....	912
25.7. Дескрипторные классы (handles).....	914
25.7.1. Операции в дескрипторных классах.....	917
25.8. Прикладные среды разработки (application frameworks).....	918
25.9. Советы.....	920
25.10. Упражнения.....	921
Приложения и предметный указатель.....	923
Приложение А. Грамматика.....	925
А.1. Введение.....	925
А.2. Ключевые слова.....	926
А.3. Лексические соглашения.....	927
А.4. Программы.....	930
А.5. Выражения.....	930
А.6. Операторы.....	934
А.7. Объявления.....	935
А.7.1. Деклараторы.....	938
А.8. Классы.....	940
А.8.1. Производные классы.....	941
А.8.2. Особые функции-члены.....	941

A8.3. Перегрузка.....	942
A. 9. Шаблоны.....	942
A.10. Обработка исключений.....	943
A. 11. Директивы препроцессора.....	944
Приложение В. Совместимость.....	947
В.1. Введение.....	947
В.2. Совместимость C и C++.....	948
В.2.1. «Тихие» отличия.....	948
В.2.2. Код на C, не являющийся C++-кодом.....	948
В.2.3. Нежелательные особенности.....	951
В.2.4. Код на C++, не являющийся кодом на C.....	951
В.3. Старые реализации C++.....	953
В.3.1. Заголовочные файлы.....	954
В.3.2. Стандартная библиотека.....	955
В.3.3. Пространства имен.....	955
В.3.4. Ошибки выделения памяти.....	956
В.3.5. Шаблоны.....	956
В.3.6. Инициализаторы в операторах for.....	958
В.4. Советы.....	958
В.5. Упражнения.....	960
Приложение С. Технические подробности.....	961
С.1. Введение и обзор.....	961
С.2. Стандарт.....	961
С.3. Символьные наборы.....	963
С.3.1. Ограниченные наборы символов.....	963
С.3.2. Escape-символы.....	964
С.3.3. Расширенные символьные наборы.....	965
С.3.4. Знаковые и беззнаковые символы.....	966
С.4. Типы целых литералов.....	967
С5. Константные выражения.....	967
С.6. Неявное преобразование типов.....	967
С.6.1. Продвижения (promotions).....	968
С.6.2. Преобразования.....	968
С.6.2.1. Интегральные преобразования.....	968
С.6.2.2. Преобразования чисел с плавающей запятой.....	969
С.6.2.3. Преобразования указателей и ссылок.....	969
С.6.2.4. Преобразования указателей на члены классов.....	970
С.6.2.5. Преобразования в логический тип.....	970
С.6.2.6. Преобразования «значение интегрального типа — значение с плавающей запятой».....	970
С.6.3. Обычные арифметические преобразования.....	971
С.7. Многомерные массивы.....	971
С.7.1. Векторы.....	971
С.7.2. Массивы.....	973
С.7.3. Передача многомерных массивов в функции.....	974
С.8. Экономия памяти.....	975
С.8.1. Битовые поля.....	976
С.8.2. Объединения.....	977
С.8.3. Объединения и классы.....	979

С.9. Управление памятью.....	979
С.9.1. Автоматическая сборка мусора.....	980
С.9.1.1. Замаскированные указатели	980
С.9.1.2. Операция delete.....	981
С.9.1.3. Деструкторы.....	982
С.9.1.4. Фрагментация памяти.....	982
С. 10. Пространства имен.	983
С. 10.1. Удобство против безопасности.....	983
С. 10.2. Вложенные пространства имен.....	984
С. 10.3. Пространства имен и классы.....	985
С.11. Управление режимами доступа.....	985
С.11.1. Доступ к членам класса.....	985
С. 11.2. Доступ к базовым классам.....	986
С.11.3. Доступ ко вложенным классам.....	988
С. 11.4. Отношение «дружбы».....	989
С. 12. Указатели на члены классов.....	989
С. 13. Шаблоны.....	990
С. 13.1. Статические члены.....	990
С.13.2. Друзья.....	991
С.13.3. Шаблоны в качестве параметров шаблонов.....	992
С. 13.4. Логический вывод аргументов функциональных шаблонов.....	992
С.13.5. Шаблоны и ключевое-слово <code>typename</code>	993
С.13.6. Ключевое слово <code>template</code> в качестве квалификатора.....	995
С. 13.7. Конкретизация.....	995
С.13.8. Связывание имен.....	996
С. 13.8.1. Зависимые имена.....	997
С.13.8.2. Связывание в точке определения.....	999
С. 13.8.3. Связывание в точке конкретизации.....	1000
С.13.8.4. Шаблоны и пространства имен.....	1001
С.13.9. Когда нужны специализации.....	1003
С.13.9.1. Конкретизация шаблона функции.....	1003
С.13.10. Явная конкретизация.....	1004
С. 14. Советы.....	1005
Приложение D. Локализация.....	1007
D.1. Национальные особенности.....	1007
D.1.1. Программирование национальных особенностей.....	1008
D.2. Класс <code>locale</code>	1011
D.2.1. Локальные контексты с заданными именами.....	1013
D.2.1.1. Конструирование новых объектов локализации.....	1015
D.2.2. Копирование и сравнение контекстов локализации.....	1017
D.2.3. Контексты <code>global()</code> и <code>classic()</code>	1018
D.2.4. Сравнение строк.....	1019
D.3. Фасеты.....	1020
D.3.1. Доступ к фасетам класса <code>locale</code>	1021
D.3.2. Простой пользовательский фасет.....	1023
D.3.3. Использование локализаций и фасетов.....	1026
D.4. Стандартные фасеты.....	1026
D.4.1. Сравнение строк.....	1029
D.4.1.1. Именованные фасеты сравнения.....	1032
D.4.2. Ввод и вывод чисел.....	1033
D.4.2.1. Пунктуация чисел.....	1033

D.4.2.2. Вывод чисел.....	1035
D.4.2.3. Ввод чисел.....	1038
D.4.3. Ввод и вывод финансовой информации.....	1039
D.4.3.1. Пунктуация денежных величин.....	1040
D.4.3.2. Вывод денежных величин.....	1043
D.4.3.3. Ввод денежных величин.....	1044
D.4.4. Ввод и вывод дат и времени.....	1046
D.4.4.1. Часы и таймеры.....	1046
D.4.4.2. Класс Date.....	1049
D.4.4.3. Вывод дат и времени.....	1049
D.4.4.4. Ввод дат и времени.....	1052
D.4.4.5. Более гибкий класс Date.....	1054
D.4.4.6. Задание формата даты.....	1056
D.4.4.7. Фасет ввода даты.....	1058
D.4.5. Классификация символов.....	1063
D.4.5.1. Вспомогательные шаблоны функций.....	1066
D.4.6. Преобразование кодов символов.....	1067
D.4.7. Сообщения.....	1071
D.4.7.1. Использование сообщений из других фасетов.....	1073
D.5. Советы.....	1075
D.6. Упражнения.....	1075
Приложение Е. Исключения и безопасность стандартной библиотеки.....	1077
E.1. Введение.....	1077
E.2. Исключения и безопасность.....	1079
E.3. Технологии реализации безопасности при исключениях.....	1083
E.3.1. Простой вектор.....	1083
E.3.2. Явное управление памятью.....	1087
E.3.3. Присваивание.....	1088
E.3.4. Метод push_back().....	1091
E.3.5. Конструкторы и инварианты.....	1093
E.3.5.1. Применение функции mit().....	1095
E.3.5.2. Полагаемся на действительное состояние по умолчанию.....	1096
E.3.5.3. Отложенное выделение ресурсов.....	1097
E.4. Гарантии стандартных контейнеров.....	1098
E.4.1. Вставка и удаление элементов.....	1099
E.4.2. Гарантии и компромиссы.....	1102
E.4.3. Функция swap().....	1105
E.4.4. Инициализация и итераторы.....	1105
E.4.5. Ссылки на элементы.....	1106
E.4.6. Предикаты.....	1107
E.5. Другие части стандартной библиотеки.....	1108
E.5.1. Строки.....	1108
E.5.2. Поток.....	1109
E.5.3. Алгоритмы.....	1109
E.5.4. Типы valarray и complex.....	1110
E.5.5. Стандартная библиотека языка C.....	1110
E.6. Рекомендации пользователям стандартной библиотеки.....	1110
E.7. Советы.....	1113
E.8. Упражнения.....	1114
Предметный указатель.....	1117