

СОДЕРЖАНИЕ

Введение	18
Для кого написана эта книга	19
Изменения в пятом издании	20
Соглашения, принятые в книге	21
От издательства	24
Глава 1. Первые шаги	25
1.1. Создание простой программы на языке C++	26
1.1.1. Компиляция и запуск программы	27
1.2. Первый взгляд на ввод-вывод	30
1.3. Несколько слов о комментариях	35
1.4. Средства управления	37
1.4.1. Оператор <code>while</code>	37
1.4.2. Оператор <code>for</code>	39
1.4.3. Ввод неизвестного количества данных	41
1.4.4. Оператор <code>if</code>	44
1.5. Введение в классы	47
1.5.1. Класс <code>Sales_item</code>	47
1.5.2. Первый взгляд на функции-члены	51
1.6. Программа для книжного магазина	52
Резюме	54
Термины	54
Часть I. Основы	59
Глава 2. Переменные и базовые типы	61
2.1. Простые встроенные типы	61
2.1.1. Арифметические типы	62
2.1.2. Преобразование типов	66
2.1.3. Литералы	70
2.2. Переменные	74
2.2.1. Определения переменных	74
2.2.2. Объявления и определения переменных	78
2.2.3. Идентификаторы	80
2.2.4. Область видимости имен	82
2.3. Составные типы	84
2.3.1. Ссылки	85
2.3.2. Указатели	86
2.3.3. Понятие описаний составных типов	92
2.4. Спецификатор <code>const</code>	95
2.4.1. Ссылка на константу	97
2.4.2. Указатели и спецификатор <code>const</code>	99

2.4.3. Спецификатор <code>const</code> верхнего уровня	101
2.4.4. Переменные <code>constexpr</code> и константные выражения	103
2.5. Работа с типами	105
2.5.1. Псевдонимы типов	106
2.5.2. Спецификатор типа <code>auto</code>	107
2.5.3. Спецификатор типа <code>decltype</code>	109
2.6. Определение собственных структур данных	111
2.6.1. Определение типа <code>Sales_data</code>	112
2.6.2. Использование класса <code>Sales_data</code>	113
2.6.3. Создание собственных файлов заголовка	116
Резюме	118
Термины	118
Глава 3. Типы <code>string</code>, <code>vector</code> и массивы	123
3.1. Пространства имен и объявления <code>using</code>	124
3.2. Библиотечный тип <code>string</code>	126
3.2.1. Определение и инициализация строк	126
3.2.2. Операции со строками	128
3.2.3. Работа с символами строки	134
3.3. Библиотечный тип <code>vector</code>	141
3.3.1. Определение и инициализация векторов	142
3.3.2. Добавление элементов в вектор	146
3.3.3. Другие операции с векторами	148
3.4. Знакомство с итераторами	153
3.4.1. Использование итераторов	153
3.4.2. Арифметические действия с итераторами	159
3.5. Массивы	161
3.5.1. Определение и инициализация встроенных массивов	162
3.5.2. Доступ к элементам массива	165
3.5.3. Указатели и массивы	167
3.5.4. Символьные строки в стиле C	172
3.5.5. Взаимодействие с устаревшим кодом	175
3.6. Многомерные массивы	177
Резюме	183
Термины	183
Глава 4. Выражения	187
4.1. Основы	188
4.1.1. Фундаментальные концепции	188
4.1.2. Приоритет и порядок	190
4.1.3. Порядок вычисления	192
4.2. Арифметические операторы	194
4.3. Логические операторы и операторы отношения	197
4.4. Операторы присвоения	201

4.5. Операторы инкремента и декремента	204
4.6. Операторы доступа к членам	207
4.7. Условный оператор	208
4.8. Побитовые операторы	210
4.9. Оператор <code>sizeof</code>	215
4.10. Оператор запятая	217
4.11. Преобразование типов	218
4.11.1. Арифметические преобразования	219
4.11.2. Другие неявные преобразования	221
4.11.3. Явные преобразования	223
4.12. Таблица приоритетов операторов	227
Резюме	229
Термины	229
Глава 5. Операторы	233
5.1. Простые операторы	233
5.2. Операторная область видимости	236
5.3. Условные операторы	236
5.3.1. Оператор <code>if</code>	237
5.3.2. Оператор <code>switch</code>	241
5.4. Итерационные операторы	247
5.4.1. Оператор <code>while</code>	247
5.4.2. Традиционный оператор <code>for</code>	248
5.4.3. Серийный оператор <code>for</code>	251
5.4.4. Оператор <code>do while</code>	253
5.5. Операторы перехода	254
5.5.1. Оператор <code>break</code>	254
5.5.2. Оператор <code>continue</code>	255
5.5.3. Оператор <code>goto</code>	256
5.6. Блоки <code>try</code> и обработка исключений	257
5.6.1. Оператор <code>throw</code>	258
5.6.2. Блок <code>try</code>	259
5.6.3. Стандартные исключения	262
Резюме	264
Термины	264
Глава 6. Функции	267
6.1. Основы функций	267
6.1.1. Локальные объекты	271
6.1.2. Объявление функций	273
6.1.3. Раздельная компиляция	274
6.2. Передача аргументов	275
6.2.1. Передача аргумента по значению	276
6.2.2. Передача аргумента по ссылке	277

6.2.3. Константные параметры и аргументы	280
6.2.4. Параметры в виде массива	283
6.2.5. Функция <code>main()</code> : обработка параметров командной строки	288
6.2.6. Функции с переменным количеством параметров	289
6.3. Типы возвращаемого значения и оператор <code>return</code>	292
6.3.1. Функции без возвращаемого значения	292
6.3.2. Функции, возвращающие значение	293
6.3.3. Возвращение указателя на массив	299
6.4. Перегруженные функции	302
6.4.1. Перегрузка и область видимости	307
6.5. Специальные средства	308
6.5.1. Аргументы по умолчанию	308
6.5.2. Встраиваемые функции и функции <code>constexpr</code>	311
6.5.3. Помощь в отладке	314
6.6. Подбор функции	317
6.6.1. Преобразование типов аргументов	320
6.7. Указатели на функции	322
Резюме	326
Термины	327
Глава 7. Классы	331
7.1. Определение абстрактных типов данных	332
7.1.1. Разработка класса <code>Sales_data</code>	332
7.1.2. Определение пересмотренного класса <code>Sales_data</code>	335
7.1.3. Определение функций, не являющихся членом класса, но связанных с ним	340
7.1.4. Конструкторы	342
7.1.5. Копирование, присвоение и удаление	347
7.2. Управление доступом и инкапсуляция	349
7.2.1. Друзья	351
7.3. Дополнительные средства класса	353
7.3.1. Снова о членах класса	353
7.3.2. Функции, возвращающие указатель <code>*this</code>	357
7.3.3. Типы классов	361
7.3.4. Снова о дружественных отношениях	363
7.4. Область видимости класса	366
7.4.1. Поиск имен в области видимости класса	367
7.5. Снова о конструкторах	372
7.5.1. Список инициализации конструктора	372
7.5.2. Делегирующий конструктор	377
7.5.3. Роль стандартного конструктора	378
7.5.4. Неявное преобразование типов класса	380
7.5.5. Агрегатные классы	384
7.5.6. Литеральные классы	385

7.6. Статические члены класса	387
Резюме	392
Термины	393
Часть II. Библиотека C++	397
Глава 8. Библиотека ввода и вывода	399
8.1. Классы ввода-вывода	400
8.1.1. Объекты ввода-вывода не допускают копирования и присвоения	401
8.1.2. Флаги состояния	402
8.1.3. Управление буфером вывода	405
8.2. Ввод и вывод в файл	407
8.2.1. Использование объектов файловых потоков	408
8.2.2. Режимы файла	411
8.3. Строковые потоки	413
8.3.1. Использование класса <code>istream</code>	414
8.3.2. Использование класса <code>ostream</code>	416
Резюме	417
Термины	417
Глава 9. Последовательные контейнеры	419
9.1. Обзор последовательных контейнеров	420
9.2. Обзор библиотечных контейнеров	423
9.2.1. Итераторы	426
9.2.2. Типы-члены классов контейнеров	429
9.2.3. Функции-члены <code>begin()</code> и <code>end()</code>	430
9.2.4. Определение и инициализация контейнера	431
9.2.5. Присвоение и функция <code>swap()</code>	435
9.2.7. Операторы сравнения	438
9.3. Операции с последовательными контейнерами	439
9.3.1. Добавление элементов в последовательный контейнер	439
9.3.2. Доступ к элементам	445
9.3.3. Удаление элементов	447
9.3.4. Специализированные функции контейнера <code>forward_list</code>	450
9.3.5. Изменение размеров контейнера	452
9.3.6. Некоторые операции с контейнерами делают итераторы недопустимыми	453
9.4. Как увеличивается размер вектора	457
9.5. Дополнительные операции со строками	462
9.5.1. Дополнительные способы создания строк	462
9.5.2. Другие способы изменения строки	464
9.5.3. Операции поиска строк	467
9.5.4. Сравнение строк	471

9.5.5. Числовые преобразования	471
9.6. Адаптеры контейнеров	473
Резюме	477
Термины	477
Глава 10. Обобщенные алгоритмы	481
10.1. Краткий обзор	482
10.2. Первый взгляд на алгоритмы	485
10.2.1. Алгоритмы только для чтения	485
10.2.2. Алгоритмы, записывающие элементы контейнера	488
10.2.3. Алгоритмы, переупорядочивающие элементы контейнера	491
10.3. Перенастройка функций	494
10.3.1. Передача функций алгоритму	494
10.3.2. Лямбда-выражения	496
10.3.3. Захват и возвращение значений лямбда-выражениями	502
10.3.4. Привязка аргументов	508
10.4. Возвращаясь к итераторам	514
10.4.1. Итераторы вставки	514
10.4.2. Поточковые итераторы	516
10.4.3. Реверсивные итераторы	521
10.5. Структура обобщенных алгоритмов	525
10.5.1. Пять категорий итераторов	525
10.5.2. Параметрическая схема алгоритмов	528
10.5.3. Соглашения об именовании алгоритмов	530
10.6. Алгоритмы, специфические для контейнеров	531
Резюме	534
Термины	534
Глава 11. Ассоциативные контейнеры	537
11.1. Использование ассоциативных контейнеров	538
11.2. Обзор ассоциативных контейнеров	541
11.2.1. Определение ассоциативного контейнера	541
11.2.2. Требования к типу ключа	543
11.2.3. Тип <code>pair</code>	545
11.3. Работа с ассоциативными контейнерами	548
11.3.1. Итераторы ассоциативных контейнеров	548
11.3.2. Добавление элементов	551
11.3.3. Удаление элементов	554
11.3.4. Индексация карт	555
11.3.5. Доступ к элементам	557
11.3.6. Карта преобразования слов	562
11.4. Неупорядоченные контейнеры	566
Использование неупорядоченного контейнера	566

Управление ячейками	567
Требования к типу ключа неупорядоченных контейнеров	568
Резюме	570
Термины	570
Глава 12. Динамическая память	573
12.1. Динамическая память и интеллектуальные указатели	574
12.1.1. Класс <code>shared_ptr</code>	575
12.1.2. Непосредственное управление памятью	584
12.1.3. Использование указателя <code>shared_ptr</code> с оператором <code>new</code>	591
12.1.4. Интеллектуальные указатели и исключения	595
12.1.5. Класс <code>unique_ptr</code>	598
12.1.6. Класс <code>weak_ptr</code>	602
12.2. Динамические массивы	606
12.2.1. Оператор <code>new</code> и массивы	607
12.2.2. Класс <code>allocator</code>	613
12.3. Использование библиотеки: программа запроса текста	617
12.3.1. Проект программы	617
12.3.2. Определение классов программы запросов	620
Резюме	625
Термины	625
Часть III. Инструменты для разработчиков классов	627
Глава 13. Управление копированием	629
13.1. Копирование, присвоение и удаление	630
13.1.1. Конструктор копий	630
13.1.2. Оператор присвоения копии	634
13.1.3. Деструктор	636
13.1.4. Правило три/пять	639
13.1.5. Использование спецификатора <code>= default</code>	642
13.1.6. Предотвращение копирования	643
13.2. Управление копированием и ресурсами	647
13.2.1. Классы, действующие как значения	648
13.2.2. Определение классов, действующих как указатели	651
13.3. Функция <code>swap()</code>	654
13.4. Пример управления копированием	658
13.5. Классы, управляющие динамической памятью	664
13.6. Перемещение объектов	673
13.6.1. Ссылки на <code>r</code> -значение	673
13.6.2. Конструктор перемещения и присваивание при перемещении	676
13.6.3. Ссылки на <code>r</code> -значение и функции-члены	690
Резюме	695
Термины	695

Глава 14. Перегрузка операторов и преобразований	699
14.1. Фундаментальные концепции	700
14.2. Операторы ввода и вывода	705
14.2.1. Перегрузка оператора вывода <<	706
14.2.2. Перегрузка оператора ввода >>	707
14.3. Арифметические операторы и операторы отношения	710
14.3.1. Операторы равенства	711
14.3.2. Операторы отношения	712
14.4. Операторы присвоения	713
14.5. Оператор индексирования	715
14.6. Операторы инкремента и декремента	717
14.7. Операторы доступа к членам	720
14.8. Оператор вызова функции	722
14.8.1. Лямбда-выражения — объекты функции	724
14.8.2. Библиотечные объекты функций	726
14.8.3. Вызываемые объекты и тип <code>function</code>	728
14.9. Перегрузка, преобразование и операторы	732
14.9.1. Операторы преобразования	733
14.9.2. Избегайте неоднозначных преобразований	737
14.9.3. Подбор функций и перегруженные операторы	742
Резюме	744
Термины	745
Глава 15. Объектно-ориентированное программирование	747
15.1. Краткий обзор ООП	748
15.2. Определение базовых и производных классов	750
15.2.1. Определение базового класса	750
15.2.2. Определение производного класса	752
15.2.3. Преобразования и наследование	759
15.3. Виртуальные функции	763
15.4. Абстрактные базовые классы	768
15.5. Управление доступом и наследование	772
15.6. Область видимости класса при наследовании	778
15.7. Конструкторы и функции управления копированием	784
15.7.1. Виртуальные деструкторы	785
15.7.2. Синтезируемые функции управления копированием и наследование	786
15.7.3. Функции-члены управления копированием производного класса	789
15.7.4. Унаследованные конструкторы	793
15.8. Контейнеры и наследование	795
15.8.1. Разработка класса <code>Basket</code>	796
15.9. Возвращаясь к запросам текста	801
15.9.1. Объектно-ориентированное решение	802

15.9.2. Классы <code>Query_base</code> и <code>Query</code>	807
15.9.3. Производные классы	809
15.9.4. Виртуальные функции <code>eval()</code>	813
Резюме	817
Термины	817
Глава 16. Шаблоны и обобщенное программирование	821
16.1. Определение шаблона	822
16.1.1. Шаблоны функций	822
16.1.2. Шаблоны класса	830
16.1.3. Параметры шаблона	841
16.1.4. Шаблоны-члены	846
16.1.5. Контроль создания экземпляра	849
16.1.6. Эффективность и гибкость	851
16.2. Дедукция аргумента шаблона	853
16.2.1. Преобразования и параметры типа шаблона	854
16.2.2. Явные аргументы шаблона функции	857
16.2.3. Замыкающие типы возвращаемого значения и трансформация типа	859
16.2.4. Указатели на функцию и дедукция аргумента	862
16.2.5. Дедукция аргумента шаблона и ссылки	863
16.2.6. Функция <code>std::move()</code>	867
16.2.7. Перенаправление	870
16.3. Перегрузка и шаблоны	873
16.4. Шаблоны с переменным количеством аргументов	878
16.4.1. Шаблоны функции с переменным количеством аргументов	880
16.4.2. Развертывание пакета	882
16.4.3. Перенаправление пакетов параметров	884
16.5. Специализация шаблона	887
Резюме	894
Термины	895
Часть IV. Дополнительные темы	897
Глава 17. Специализированные средства библиотек	899
17.1. Тип <code>tuple</code>	899
17.1.1. Определение и инициализация кортежей	901
17.1.2. Использование кортежей для возвращения нескольких значений	903
17.2. Тип <code>bitset</code>	906
17.2.1. Определение и инициализация наборов битов	906
17.2.2. Операции с наборами битов	909

17.3. Регулярные выражения	912
17.3.1. Использование библиотеки регулярных выражений	913
17.3.2. Типы итераторов классов соответствия и <code>regex</code>	920
17.3.3. Использование подвыражений	925
17.3.4. Использование функции <code>regex_replace()</code>	929
17.4. Случайные числа	933
17.4.1. Процессоры случайных чисел и распределения	934
17.4.2. Другие виды распределений	939
17.5. Еще о библиотеке ввода и вывода	943
17.5.1. Форматированный ввод и вывод	943
17.5.2. Не форматированные операции ввода-вывода	952
17.5.3. Произвольный доступ к потоку	956
Резюме	961
Термины	962
Глава 18. Инструменты для крупномасштабных программ	965
18.1. Обработка исключений	966
18.1.1. Передача исключений	966
18.1.2. Обработка исключения	970
18.1.3. Блок <code>try</code> функции и конструкторы	973
18.1.4. Спецификатор исключения <code>noexcept</code>	974
18.1.5. Иерархии классов исключений	978
18.2. Пространства имен	981
18.2.1. Определение пространств имен	981
18.2.2. Использование членов пространства имен	990
18.2.3. Классы, пространства имен и области видимости	995
18.2.4. Перегрузка и пространства имен	999
18.3. Множественное и виртуальное наследование	1002
18.3.1. Множественное наследование	1003
18.3.2. Преобразования и несколько базовых классов	1006
18.3.3. Область видимости класса при множественном наследовании	1009
18.3.4. Виртуальное наследование	1011
18.3.5. Конструкторы и виртуальное наследование	1015
Резюме	1018
Термины	1018
Глава 19. Специализированные инструменты и технологии	1023
19.1. Контроль распределения памяти	1023
19.1.1. Перегрузка операторов <code>new</code> и <code>delete</code>	1024
19.1.2. Размещающий оператор <code>new</code>	1028
19.2. Идентификация типов времени выполнения	1029
19.2.1. Оператор <code>dynamic_cast</code>	1030
19.2.2. Оператор <code>typeid</code>	1032

19.2.3. Использование RTTI	1034
19.2.4. Класс <code>type_info</code>	1037
19.3. Перечисления	1038
19.4. Указатель на член класса	1042
19.4.1. Указатели на переменные-члены	1043
19.4.2. Указатели на функции-члены	1045
19.4.3. Использование функций-членов как вызываемых объектов	1049
19.5. Вложенные классы	1052
19.6. Класс объединения, экономящий место	1056
19.7. Локальные классы	1063
19.8. Возможности, снижающие переносимость	1065
19.8.1. Битовые поля	1065
19.8.2. Спецификатор <code>volatile</code>	1067
19.8.3. Директивы компоновки: <code>extern "C"</code>	1068
Резюме	1072
Термины	1073
Приложение А. Библиотека	1077
А.1. Имена и заголовки стандартной библиотеки	1077
А.2. Краткий обзор алгоритмов	1082
А.2.1. Алгоритмы поиска объекта	1083
А.2.2. Другие алгоритмы, осуществляющие только чтение	1084
А.2.3. Алгоритмы бинарного поиска	1085
А.2.4. Алгоритмы записи в элементы контейнера	1086
А.2.5. Алгоритмы сортировки и разделения	1089
А.2.6. Общие функции изменения порядка	1091
А.2.7. Алгоритмы перестановки	1093
А.2.8. Алгоритмы набора для отсортированных последовательностей	1094
А.2.9. Минимальные и максимальные значения	1095
А.2.10. Числовые алгоритмы	1097
А.3. Случайные числа	1098
А.3.1. Распределение случайных чисел	1098
А.3.2. Процессоры случайных чисел	1100
Предметный указатель	1103