

Содержание

Введение	16
О чем эта книга	16
Прилагаемый компакт-диск	16
Что такое C++	17
Соглашения, используемые в книге	17
Как организована эта книга	18
И еще...	18
Часть I. Первое знакомство с C++	18
Часть II. Становимся функциональными программистами	18
Часть III. Введение в классы	18
Часть IV. Наследование	19
Часть V. Полезные особенности	19
Часть VI. “Великолепные десятки”	19
Использованные в этой книге пиктограммы	19
Что дальше	20
ЧАСТЬ I. ПЕРВОЕ ЗНАКОМСТВО С C++	21
Глава 1. Написание вашей первой программы	23
Постигая концепции C++	23
Установка Code::Blocks	24
Создание первой программы	26
Создание проекта	27
Ввод кода	28
Как облегчить жизнь	30
Построение программы	31
Выполнение программы	31
Разбор программ	33
Определение структуры программ C++	33
Использование в исходном коде комментариев	33
Использование инструкций в программах	34
Объявления	34
Генерация вывода	35
Вычисление выражений	35
Сохранение результатов выражения	36
Обзор программы Convert продолжается...	36
Глава 2. Премудрости объявления переменных	37
Объявление переменных	37
Объявление разных типов переменных	38
Ограничения, налагаемые на целые числа в C++	39
Решение проблемы усечения дробной части	39
Ограничения чисел с плавающей точкой	40
Объявления типов переменных	41
Типы констант	43

Диапазоны числовых типов	44
Специальные символы	44
Широкая дорога для символов	45
Логические выражения	46
Выражения смешанного типа	46
Автоматические объявления	47
Глава 3. Выполнение математических операций	49
Бинарная арифметика	49
Анализ выражений	50
Определение порядка операций	51
Выполнение унарных операций	51
Использование операторов присвоения	52
Глава 4. Выполнение логических операций	55
Зачем нужны логические операторы	55
Использование простых логических операторов	55
Хранение логических значений	56
Использование целых переменных в качестве логических	57
Логические операции с действительными переменными	58
Бинарные числа в C++	59
Десятичная система счисления	59
Другие системы счисления	60
Двоичная система счисления	60
Выполнение побитовых логических операций	61
Побитовые операции с одним битом	62
Использование побитовых операторов	63
Простой пример	63
Практическое применение логических вычислений	65
Глава 5. Управление потоком выполнения программы	67
Управление ходом программы с помощью команд ветвления	67
Выполнение циклов	69
Цикл while	69
Использование операторов инкремента и декремента	71
Использование цикла for	72
Избегайте бесконечных циклов	74
Специальные операторы управления циклом	74
Вложенные команды управления	77
Инструкция выбора	78
ЧАСТЬ II. СТАНОВИМСЯ ФУНКЦИОНАЛЬНЫМИ ПРОГРАММИСТАМИ	81
Глава 6. Создание функций	83
Написание и использование функций	83
Определение первой функции	85
Определение функции sumSequence()	85
Вызов функции sumSequence()	86
Разделяй и властвуй	86
Подробный анализ функций	86
Простые функции	87
Функции с аргументами	87

Перегрузка функций	90
Определение прототипов функций	91
Хранение переменных в памяти	92
Глава 7. Хранение последовательностей в массивах	95
Преимущества массивов	95
Работа с массивами	96
Инициализация массива	99
Выход за границы массива	100
Использовать ли массивы	100
Определение и использование массивов с элементами-массивами	100
Использование символьных массивов	101
Создание массива символов	101
Создание строки символов	102
Управление строками	103
Библиотечные функции	105
Работа с широкими строками	106
Глава 8. Первое знакомство с указателями в C++	109
Размер переменной	109
Что такое адрес	110
Адресные операторы	111
Использование указателей	112
Использование разных типов указателей	113
Передача указателей функциям	114
Передача аргументов по значению	114
Передача значений указателей	115
Передача аргументов по ссылке	115
Постоянные проблемы с const	116
Использование кучи	117
Область видимости	117
Проблемы области видимости	118
Решение с применением кучи	119
Глава 9. Второе знакомство с указателями	121
Операции с указателями	121
Повторное знакомство с массивами в свете указателей	122
Операции над указателями для адресации внутри массива	123
Указатели и работа со строками	124
Почему при работе со строками пользуются указателями	125
Операции с указателями других типов	126
Отличия между указателями и массивами	126
Когда указатель никуда не указывает	128
Объявление и использование массивов указателей	129
Использование массивов строк	129
Доступ к аргументам main()	131
Глава 10. Препроцессор C++	135
Что такое препроцессор	135
Включение файлов	136
Директива #define	138

Как обойтись без #define	140
Перечисления	141
Использование #if	142
Предопределенные константы	143
Typedef	145
ЧАСТЬ III. ВВЕДЕНИЕ В КЛАССЫ	147
Глава 11. Знакомство с объектно-ориентированным программированием	149
Микроволновые печи и уровни абстракции	149
Приготовление блюд с помощью функций	150
Приготовление “объектно-ориентированных” блюд	150
Классификация микроволновых печей	150
Зачем нужна классификация	151
Глава 12. Классы в C++	153
Введение в классы	153
Формат класса	153
Обращение к членам класса	154
Активация наших объектов	155
Моделирование реальных объектов	155
Зачем нужны функции-члены	156
Добавление функции-члена	156
Вызов функций-членов	157
Доступ к членам из функции-члена	158
Разрешение области видимости	160
Определение функции-члена	160
Определение функций-членов вне класса	162
Перегрузка функций-членов	163
Глава 13. Указатели на объекты	165
Объявление массивов объектов	165
Объявление указателей на объекты	166
Разыменование указателей на объекты	167
Использование стрелок	167
Передача объектов функциям	168
Вызов функции с передачей объекта по значению	168
Вызов функции с передачей указателя на объект	169
Передача объекта по ссылке	170
Зачем использовать указатели и ссылки	171
Возврат к куче	171
Сравнение указателей и ссылок	173
Использование связанных списков	173
Другие операции над связанным списком	174
Программа LinkedListData	174
Списки в стандартной библиотеке	177
Глава 14. Защищенные члены класса: не беспокоить!	179
Защищенные члены	179
Зачем нужны защищенные члены	179
Как устроены защищенные члены	180
Чем хороши защищенные члены	181

Защита внутреннего устройства класса	181
Классы с ограниченным интерфейсом	182
Обращение внешних функций к защищенным членам	182
Глава 15. Создание и удаление объектов	185
Создание объектов	185
Использование конструкторов	186
Конструирование одного объекта	186
Конструирование нескольких объектов	187
Конструирование составных объектов	188
Что такое деструктор	190
Зачем нужен деструктор	190
Работа с деструкторами	190
Глава 16. Аргументация конструирования	195
Как снабдить конструктор аргументами	195
Как использовать конструктор с аргументами	195
Перегрузка конструктора	197
Определение конструкторов по умолчанию	199
Конструирование членов класса	201
Конструкторы константных членов	204
Управление последовательностью конструирования	204
Локальные объекты создаются последовательно	205
Статические объекты создаются один раз	205
Все глобальные объекты создаются до вызова main()	206
Порядок создания глобальных объектов не определен	206
Члены создаются в порядке их объявления	207
Деструкторы удаляют объекты в порядке, обратном порядку их создания	208
Конструкторы как разновидность преобразования типов	208
Глава 17. Копирующий конструктор	209
Копирование объекта	209
Зачем нужен копирующий конструктор	209
Использование копирующего конструктора	210
Автоматический копирующий конструктор	211
“Мелкие” и “глубокие” копии	213
Временные объекты	216
Как избегать временных объектов	217
Глава 18. Статические члены	219
Определение статических членов	219
Зачем нужны статические члены	219
Использование статических членов	220
Обращение к статическим данным-членам	221
Применение статических данных-членов	222
Объявление статических функций-членов	222
Что такое this	224
ЧАСТЬ IV. НАСЛЕДОВАНИЕ	227
Глава 19. Наследование классов	229
Зачем нужно наследование	230
Как наследуется класс	230

Использование подкласса	232
Конструирование подкласса	232
Деструкция подкласса	233
Отношение СОДЕРЖИТ	234
Глава 20. Знакомство с виртуальными функциями-членами: настоящие ли они	235
Зачем нужен полиморфизм	237
Как работает полиморфизм	238
Когда функция не будет виртуальной	239
Виртуальные особенности	240
Глава 21. Разложение классов	243
Разложение	243
Реализация абстрактных классов	246
Концепция абстрактных классов	247
Создание полноценного класса из абстрактного	248
Передача абстрактных классов	248
ЧАСТЬ V. ПОЛЕЗНЫЕ ОСОБЕННОСТИ	251
Глава 22. Оператор присваивания	253
Сравнение операторов и функций	253
Потоковые операторы	254
Мелкое копирование — глубокие проблемы	254
Переопределение оператора присваивания	255
Перегрузка оператора индексирования	259
Глава 23. Использование потоков ввода-вывода	261
Как работают потоки ввода-вывода	261
Потоковые объекты по умолчанию	262
Потоковый ввод-вывод	263
Режимы открытия файлов	264
Состояние файла	264
Небольшой пример	265
Методы классов потоков	267
Прямое чтение и запись потока	269
Управление форматированием	270
Что такое endl	272
Позиционирование в файле	272
Подклассы stringstream	272
Работа с манипуляторами	275
Глава 24. Обработка ошибок и исключения	277
Зачем нужен новый механизм обработки ошибок	278
Механизм исключительных ситуаций	279
Что же мы будем бросать?	282
Передай дальше	284
Глава 25. Множественное наследование	285
Механизм множественного наследования	285
Устранение неоднозначностей множественного наследования	287
Виртуальное наследование	287

Конструирование объектов	293
Отрицательные стороны множественного наследования	293
Глава 26. Шаблоны C++	295
Обобщение функции в шаблон	296
Шаблоны классов	297
Советы по использованию шаблонов	300
Глава 27. Стандартная библиотека шаблонов	301
Контейнер string	301
Контейнер list	305
Огласите весь список, пожалуйста!	307
Операции над всем списком	308
Приведите пример	308
ЧАСТЬ VI. ВЕЛИКОЛЕПНЫЕ ДЕСЯТКИ	311
Глава 28. Десять способов избежать ошибок	313
Включение всех предупреждений и сообщений об ошибках	313
Используйте последовательный стиль программирования	313
Ограничивайте видимость	314
Комментируйте свою программу	315
Хотя бы один раз выполните программу пошагово	315
Избегайте перегрузки операторов	316
Работа с кучей	316
Используйте для обработки ошибок исключительные ситуации	316
Делайте деструкторы виртуальными	316
Избегайте множественного наследования	318
Глава 29. Десять наиболее важных добавлений в C++	319
Использование интеллектуальных указателей	319
Инициализация переменных списками переменной длины	320
Встроенная инициализация членов-данных	321
Инстанцирование внешних шаблонов	321
Локальная память потока	322
Применение ссылок rvalue	322
Реализация концепций	323
Лямбда-выражения	324
Определение вариативных шаблонов	324
Применение typeid()	324
Приложение. О компакт-диске	327
Системные требования	327
Использование компакт-диска	327
Что есть на компакт-диске	328
Программы .CPP	328
Интегрированная среда разработки Code::Blocks	329
Решение проблем	329
Предметный указатель	330