

Содержание

Об авторе	19
Посвящается	19
Благодарности	20
От издательства	21
Введение	23
Предисловие к пятому изданию	23
Особенности данного учебника	23
Примеры кода, приведенные в этой книге	24
Об этой книге	24
Примечание для преподавателей	29
Соглашения, используемые в этой книге	30
Системы, использованные при разработке примеров для данной книги	31
Глава 1. С чего начать?	33
Изучение языка C++: с чем вы будете иметь дело	34
Истоки языка C++: немного истории	34
Язык программирования C	35
Философия программирования на языке C	36
Переход к C++: объектно-ориентированное программирование	37
C++ и обобщенное программирование	39
Происхождение языка программирования C++	39
Переносимость и стандарты	40
Порядок создания программы	43
Создание файла исходного кода	44
Компиляция и компоновка	46
Компиляция и связывание в Unix	46
Компиляция и связывание в Linux	47
Компиляторы командной строки для MS-DOS	48
Компиляторы для Windows	48
C++ в компьютерах Macintosh	51
Резюме	51
Глава 2. Приступаем к изучению C++	53
Первые шаги в C++	53
Функция main()	55
Заголовок функции как интерфейс	56
Почему именно main()?	58
Комментарии в языке C++	58
Препроцессор C++ и файл iostream	59
Имена заголовочных файлов	60

Пространства имен	61
Вывод в C++ с помощью <code>cout</code>	63
Манипулятор <code>endl</code>	64
Символ новой строки	65
Форматирование исходного кода C++	65
Лексемы и обобщенный пробел	66
Стиль написания исходного кода C++	66
Операторы в языке C++	67
Операторы объявления и переменные	68
Операторы присваивания	70
Новый трюк с объектом <code>cout</code>	70
Другие операторы C++	71
Использование <code>cin</code>	72
Конкатенация с помощью <code>cout</code>	73
<code>cin</code> и <code>cout</code> : признак класса	73
Функции	75
Использование функции, имеющей возвращаемое значение	75
Разновидности функций	79
Функции, определяемые пользователем	80
Использование определяемых пользователем функций, имеющих возвращаемое значение	83
Местоположение директивы <code>using</code> в программах со множеством функций	85
Резюме	87
Вопросы для самоконтроля	88
Упражнения по программированию	89
Глава 3. Работа с данными	91
Простые переменные	92
Имена, присваиваемые переменным	92
Целочисленные типы	94
Целочисленные типы <code>short</code> , <code>int</code> и <code>long</code>	95
Типы без знаков	100
Выбор целочисленного типа	102
Целочисленные константы	103
Как компилятор C++ определяет тип константы	105
Тип <code>char</code> : символы и короткие целые числа	106
Тип <code>bool</code>	114
Квалификатор <code>const</code>	115
Числа с плавающей точкой	116
Запись чисел с плавающей точкой	117
Типы чисел с плавающей точкой	118
Константы с плавающей точкой	121
Преимущества и недостатки чисел с плавающей точкой	121
Арифметические операции в языке C++	122
Порядок выполнения операций: приоритеты операций и ассоциативность	124
Операция нахождения остатка целочисленного деления	127
Преобразования типов	128

8 Содержание

Резюме	133
Вопросы для самоконтроля	134
Упражнения по программированию	135
Глава 4. Составные типы	137
Введение в массивы	138
Замечания по программе	140
Правила инициализации массивов	141
Строки	142
Конкатенация строковых констант	144
Использование строк в массивах	144
Риск, связанный с вводом строк	146
Построчное чтение ввода	147
Смешивание строкового и числового ввода	151
Введение в класс <code>string</code>	153
Присваивание, конкатенация и добавление	154
Дополнительные сведения об операциях над классом <code>string</code>	155
Дополнительные сведения о строковом вводе-выводе	157
Введение в структуры	159
Использование структур в программах	161
Может ли структура содержать член типа <code>string?</code>	163
Прочие свойства структур	164
Массивы структур	165
Битовые поля в структурах	167
Объединения	167
Перечисления	169
Установка значений перечислителей	171
Диапазоны значений перечислителей	171
Указатели и свободное хранилище	172
Объявление и инициализация указателей	175
Опасность указателей	177
Указатели и числа	178
Выделение памяти операцией <code>new</code>	178
Освобождение памяти операцией <code>delete</code>	181
Использование <code>new</code> для создания динамических массивов	182
Указатели, массивы и арифметика указателей	185
Замечания по программе	186
Указатели и строки	190
Использование <code>new</code> для создания динамических структур	195
Автоматическое, статическое и динамическое хранилища	198
Резюме	200
Вопросы для самоконтроля	201
Упражнения по программированию	202
Глава 5. Циклы и выражения отношений	205
Введение в цикл <code>for</code>	205
Части цикла <code>for</code>	207

Вернемся к циклу <code>for</code>	213
Изменение шага цикла	215
Внутри строки с помощью цикла <code>for</code>	216
Операции инкремента и декремента	217
Побочные эффекты и последовательные точки	218
Сравнение префиксной и постфиксной форм	219
Операции инкремента и декремента и указатели	220
Комбинация операций присваивания	221
Составные операторы, или блоки	221
Операция запятой (или еще о синтаксических трюках)	223
Выражения отношений	226
Вероятные ошибки	227
Сравнение строк в стиле C	229
Сравнение строк – объектов класса <code>string</code>	232
Цикл <code>while</code>	233
Замечания по программе	235
Сравнение циклов <code>for</code> и <code>while</code>	235
Подождем минуточку – построение цикла задержки	237
Цикл <code>do while</code>	239
Циклы и текстовый ввод	242
Применение для ввода простого <code>cin</code>	242
Спасение в виде <code>cin.get(char)</code>	243
Который из <code>cin.get()</code> ?	244
Условие конца файла	245
Еще одна версия <code>cin.get()</code>	248
Вложенные циклы и двумерные массивы	252
Инициализация двумерного массива	253
Резюме	255
Вопросы для самоконтроля	256
Упражнения по программированию	257
Глава 6. Операторы ветвления и логические операции	259
Оператор <code>if</code>	259
Оператор <code>if else</code>	261
Форматирование операторов <code>if else</code>	263
Конструкция <code>if else if else</code>	264
Логические выражения	266
Логическая операция ИЛИ: <code> </code>	266
Логическая операция И: <code>&&</code>	267
Логическая операция НЕ: <code>!</code>	272
Факты о логических операциях	273
Альтернативные представления	275
Библиотека символьных функций <code>cctype</code>	275
Операция <code>?:</code>	277
Оператор <code>switch</code>	279
Использование перечислителей в качестве меток	282
<code>switch</code> и <code>if else</code>	283

10 Содержание

Операторы <code>break</code> и <code>continue</code>	284
Замечания по программе	285
Циклы для чтения чисел	286
Замечания по программе	289
Простой файловый ввод-вывод	290
Текстовый ввод-вывод и текстовые файлы	290
Запись текстового файла	292
Чтение текстового файла	295
Резюме	301
Вопросы для самоконтроля	301
Упражнения по программированию	303

Глава 7. Функции: программные модули C++ **307**

Обзор функций	307
Определение функции	308
Прототипирование и вызов функции	311
Аргументы функций и передача по значению	314
Множественные аргументы	315
Еще одна функция с двумя аргументами	318
Функции и массивы	320
Как указатели позволяют функциям обрабатывать массивы	322
Последствия использования массивов в качестве аргументов	323
Дополнительные примеры функций с массивами	325
Функции, работающие с диапазонами массивов	331
Указатели и <code>const</code>	332
Функции и двумерные массивы	336
Функции и строки в стиле C	338
Функции с аргументами — строками в стиле C	338
Функции, возвращающие строки в стиле C	340
Функции и структуры	341
Передача и возврат структур	342
Еще один пример использования функций со структурами	344
Передача адресов структур	348
Функции и объекты класса <code>string</code>	350
Рекурсия	352
Рекурсия с одиночным рекурсивным вызовом	352
Рекурсия с множественными рекурсивными вызовами	353
Указатели на функции	355
Основы указателей на функции	355
Пример с указателем на функцию	358
Резюме	359
Вопросы для самоконтроля	360
Упражнения по программированию	361

Глава 8. Дополнительные сведения о функциях **365**

Встроенные функции C++	365
Ссылочные переменные	368

Создание ссылочных переменных	369
Ссылки в роли параметров функций	372
Свойства и особенности ссылок	375
Временные переменные, ссылочные аргументы и спецификатор <code>const</code>	377
Использование ссылок при работе со структурами	379
Использование ссылок на объект класса	383
Еще один урок ООП: объекты, наследование и ссылки	387
Когда целесообразно использовать ссылочные аргументы	390
Аргументы, определяемые по умолчанию	391
Перегрузка функций	394
Пример перегрузки	396
Когда целесообразно использовать перегрузку функций	399
Шаблоны функций	400
Перегруженные шаблоны	403
Явные специализации	405
Создание экземпляров и специализация	409
Какую версию функции выбирает компилятор?	411
Резюме	417
Вопросы для самоконтроля	418
Упражнения по программированию	419
Глава 9. Модели памяти и пространства имен	421
Раздельная компиляция	421
Продолжительность существования области хранения, область видимости и компоновка	427
Область видимости и связывание	428
Автоматическая продолжительность хранения	428
Статическая продолжительность хранения	434
Спецификаторы и классификаторы	445
Функции и связывание	447
Языковое связывание	448
Схемы хранения и динамическое распределение памяти	449
Операция <code>new</code> с адресацией	450
Пространства имен	454
Традиционные пространства имен C++	454
Новые свойства пространства имен	456
Пример пространства имен	463
Будущее пространств имен	467
Резюме	468
Вопросы для самоконтроля	469
Упражнения по программированию	471
Глава 10. Объекты и классы	475
Процедурное и объектно-ориентированное программирование	476
Абстракции и классы	477
Что такое тип?	477
Классы в C++	478

12 Содержание

Реализация функций-членов класса	483
Использование классов	488
Обзор	492
Конструкторы и деструкторы классов	493
Усовершенствование класса <code>Stock</code>	499
Обзор конструкторов и деструкторов	506
Изучение объектов: указатель <code>this</code>	507
Массив объектов	513
Возврат к интерфейсу и реализации	516
Область видимости класса	517
Константы области видимости класса	518
Абстрактные типы данных	519
Резюме	524
Вопросы для самоконтроля	525
Упражнения по программированию	526
Глава 11. Работа с классами	529
Перегрузка операций	530
Время в наших руках: разработка примера перегрузки операции	531
Добавление операции сложения	534
Ограничения перегрузки	537
Дополнительные сведения о перегруженных операциях	539
Что такое друзья?	541
Создание друзей	543
Общий вид друга: перегрузка операции <code><<</code>	544
Перегруженные операции: сравнение функций-членов и функций-не-членов	550
Дополнительные сведения о перегрузке: класс <code>Vector</code>	551
Использование члена состояния	559
Перегрузка арифметических операций для класса <code>Vector</code>	561
Комментарии к реализации	563
Использование класса <code>Vector</code> в программе “Случайная прогулка”	564
Автоматическое преобразование и приведение типов в классах	567
Преобразования и друзья	578
Резюме	582
Вопросы для самоконтроля	583
Упражнения по программированию	584
Глава 12. Классы и динамическое распределение памяти	587
Динамическая память и классы	588
Простой пример и статические члены класса	588
Неявные функции-члены	597
Новый усовершенствованный класс <code>String</code>	606
О чем следует помнить при использовании операции <code>new</code> в конструкторах	617
Замечания о возвращаемых объектах	619
Использование указателей на объекты	623
Обзор технических приемов	633

Моделирование очереди	635
Класс <code>Queue</code>	635
Класс <code>Customer</code>	646
Моделирование	649
Резюме	654
Вопросы для самоконтроля	655
Упражнения по программированию	657

Глава 13. Наследование классов 661

Начало работы с простым базовым классом	662
Порождение класса	664
Конструкторы: анализ доступа	666
Использование производного класса	669
Специальные отношения между производным и базовым классами	671
Наследование: отношение is-a	673
Полиморфное общедоступное наследование	675
Разработка классов <code>Brass</code> и <code>BrassPlus</code>	676
Статическое и динамическое связывание	688
Совместимость типов указателя и ссылки	688
Виртуальные методы и динамическое связывание	689
Что следует знать о виртуальных методах	693
Управление доступом: <code>protected</code>	696
Абстрактные базовые классы	698
Использование концепции АБК	701
Философия АБК	705
Наследование и динамическое распределение памяти	705
Случай 1: производный класс не использует операцию <code>new</code>	706
Случай 2: производный класс не использует операцию <code>new</code>	707
Пример наследования с динамическим распределением памяти и друзьями	709
Обзор проекта класса	714
Функции-члены, которые генерирует компилятор	714
Анализ других методов класса	716
Анализ общедоступного наследования	719
Сводка функций классов	724
Резюме	725
Вопросы для самоконтроля	726
Упражнения по программированию	727

Глава 14. Повторное использование кода в C++ 731

Классы с членами-объектами	732
Класс <code>valarray</code> : краткий обзор	732
Проект класса <code>Student</code>	733
Пример класса <code>Student</code>	735
Инициализация включенных объектов	737
Использование интерфейса для включенного объекта	738
Использование нового класса <code>Student</code>	740
Приватное наследование	742

14 Содержание

Пример класса Student (новая версия)	743
Инициализация компонентов базового класса	743
Множественное наследование	753
Сколько всего сотрудников?	758
Какой использовать метод?	761
Краткий обзор множественного наследования	771
Шаблоны класса	772
Определение шаблона класса	772
Использование шаблонного класса	776
Более пристальный взгляд на шаблонные классы	778
Пример шаблона массива и нетипизированные аргументы	784
Универсальность шаблонов	786
Рекурсивное использование шаблонов	786
Специализация шаблона	789
Члены-шаблоны	792
Шаблоны как параметры	795
Шаблонные классы и друзья	797
Резюме	804
Вопросы для самоконтроля	806
Упражнения по программированию	808
Глава 15. Дружественность, исключения и другие понятия	813
Друзья	813
Дружественные классы	814
Дружественные функции-члены	818
Другие дружественные отношения	821
Вложенные классы	823
Вложенные классы и доступ	825
Вложение в шаблон	827
Исключения	830
Вызов abort ()	831
Возврат кода ошибки	832
Механизм исключений	834
Использование объектов в качестве исключений	837
Раскручивание стека	841
Дополнительные свойства исключений	847
Класс exception	849
Исключения, классы и наследование	853
Потеря исключений	858
Предостережения при использовании исключений	861
RTTI	863
Для чего нужен RTTI	863
Как работает RTTI?	864
Операция dynamic_cast	864
Операции приведения типов	872
Резюме	875
Вопросы для самоконтроля	876
Упражнения по программированию	877

Глава 16. Класс <code>string</code> и стандартная библиотека шаблонов	879
Класс <code>string</code>	879
Создание объекта <code>string</code>	880
Ввод для класса <code>string</code>	884
Работа со строками	886
Дополнительные возможности класса <code>string</code>	891
Класс <code>auto_ptr</code>	894
Использование <code>auto_ptr</code>	895
Соображения по использованию <code>auto_ptr</code>	897
Стандартная библиотека шаблонов (STL)	899
Шаблон <code>vector</code>	899
Что еще можно делать с помощью векторов	901
Дополнительные возможности векторов	906
Обобщенное программирование	910
Зачем нужны итераторы?	910
Виды итераторов	915
Иерархия итераторов	917
Концепции, уточнения и модели	919
Виды контейнеров	926
Ассоциативные контейнеры	936
Функциональные объекты (также известные как функторы)	942
Концепции функторов	943
Предопределенные функторы	946
Адаптируемые функторы и функциональные адаптеры	948
Алгоритмы	950
Группы алгоритмов	950
Основные свойства алгоритмов	951
STL и класс <code>string</code>	952
Сравнение функций и методов контейнеров	954
Использование STL	955
Другие библиотеки	959
<code>vector</code> и <code>valarray</code>	959
Резюме	965
Вопросы для самоконтроля	967
Упражнения по программированию	968
Глава 17. Ввод, вывод и файлы	971
Обзор ввода и вывода в C++	972
Потоки и буферы	972
Потоки, буферы и файл <code>iostream</code>	975
Перенаправление	977
Вывод с помощью <code>cout</code>	978
Перегруженная операция <code><<</code>	979
Другие методы <code>ostream</code>	982
Сброс содержимого выходного буфера	984
Форматирование с помощью <code>cout</code>	985
Ввод с помощью <code>cin</code>	1001

16 Содержание

Как <code>cin >></code> воспринимает ввод	1003
Состояния потока	1005
Другие методы класса <code>istream</code>	1010
Другие методы <code>istream</code>	1017
Файловый ввод и вывод	1021
Простой файловый ввод-вывод	1021
Проверка потока и <code>is_open()</code>	1024
Открытие нескольких файлов	1025
Обработка командной строки	1026
Режимы файла	1028
Произвольный доступ	1039
Внутреннее форматирование	1046
Что теперь?	1049
Резюме	1050
Вопросы для самоконтроля	1051
Упражнения по программированию	1052
Приложение А. Основания систем счисления	1057
Десятичные числа (основание 10)	1057
Восьмеричные целые числа (основание 8)	1057
Шестнадцатеричные числа (основание 16)	1058
Двоичные числа (основание 2)	1058
Двоичная и шестнадцатеричная формы записи	1059
Приложение Б. Зарезервированные слова языка C++	1061
Служебные слова C++	1061
Альтернативные лексемы	1062
Зарезервированные имена библиотек C++	1062
Приложение В. Набор символов ASCII	1065
Приложение Г. Приоритеты операций	1069
Приложение Д. Другие операции	1073
Битовые операции	1073
Операции сдвига	1073
Логические битовые операции	1075
Альтернативные варианты представления битовых операций	1077
Примеры использования битовых операций	1078
Операции разыменования членов	1079
Приложение Е. Шаблонный класс <code>string</code>	1085
Тринадцать типов и константа	1085
Информация о данных, конструкторы и вспомогательные элементы	1087
Конструкторы по умолчанию	1089

Конструкторы, использующие массивы	1089
Конструкторы, использующие часть массива	1090
Конструкторы копирования	1090
Конструкторы, использующие n копий символа	1091
Конструкторы, использующие диапазон	1092
Различные операции с памятью	1092
Доступ к строке	1093
Основные варианты присваивания	1094
Поиск строки	1094
Семейство <code>find()</code>	1094
Семейство <code>rfind()</code>	1095
Семейство <code>find_first_of()</code>	1095
Семейство <code>find_last_of()</code>	1096
Семейство <code>find_first_not_of()</code>	1096
Семейство <code>find_last_not_of()</code>	1097
Методы и функции сравнения	1097
Модификаторы строк	1098
Методы присоединения и добавления	1098
Дополнительные методы присваивания	1099
Методы вставки	1100
Методы удаления	1100
Методы замены	1101
Другие методы модифицирования: <code>copy()</code> и <code>swap()</code>	1102
Вывод и ввод	1102

Приложение Ж. Методы и функции библиотеки STL 1103

Члены, общие для всех контейнеров	1103
Дополнительные члены для векторов, списков и двусторонних очередей	1106
Дополнительные члены для множеств и таблиц	1108
Функции библиотеки STL	1110
Операции, не изменяющие последовательности	1110
Операции, видоизменяющие последовательности	1115
Операции сортировки и связанные с ними операции	1123
Сортировка	1125
Бинарный поиск	1127
Слияние	1128
Работа с множествами	1129
Работа с частично упорядоченными полными бинарными деревьями	1131
Поиск максимального и минимального значений	1132
Работа с перестановками	1134
Числовые операции	1135

Приложение З. Рекомендуемая литература и ресурсы в Internet 1139

Рекомендуемая литература	1139
Ресурсы в Internet	1141

Приложение И. Переход к стандарту ANSI/ISO C++	1143
Используйте альтернативные варианты для некоторых директив препроцессора	1143
Используйте <code>const</code> вместо <code>#define</code> для определения констант	1143
Используйте <code>inline</code> вместо <code>#define</code> для определения коротких функций	1145
Используйте прототипы функций	1146
Используйте приведение типов	1146
Знакомьтесь с функциональными особенностями C++	1147
Используйте новую организацию заголовков	1147
Используйте пространства имен	1148
Используйте шаблон <code>auto_ptr</code>	1149
Используйте класс <code>string</code>	1149
Используйте библиотеку STL	1150
Приложение К. Ответы на вопросы для самоконтроля	1151
Ответы на вопросы для самоконтроля из главы 2	1151
Ответы на вопросы для самоконтроля из главы 3	1152
Ответы на вопросы для самоконтроля из главы 4	1153
Ответы на вопросы для самоконтроля из главы 5	1154
Ответы на вопросы для самоконтроля из главы 6	1155
Ответы на вопросы для самоконтроля из главы 7	1156
Ответы на вопросы для самоконтроля из главы 8	1158
Ответы на вопросы для самоконтроля из главы 9	1160
Ответы на вопросы для самоконтроля из главы 10	1161
Ответы на вопросы для самоконтроля из главы 11	1163
Ответы на вопросы для самоконтроля из главы 12	1164
Ответы на вопросы для самоконтроля из главы 13	1167
Ответы на вопросы для самоконтроля из главы 14	1168
Ответы на вопросы для самоконтроля из главы 15	1170
Ответы на вопросы для самоконтроля из главы 16	1171
Ответы на вопросы для самоконтроля из главы 17	1172
Предметный указатель	1175