

Оглавление

Предисловие от издательства	20
Об авторе	21
О рецензентах	22
Вступительное слово автора	23
Вступительное слово от сообщества C++	24
Предисловие	25
Глава 1. Приступаем к написанию приложения	29
Вступление.....	29
Получение параметров конфигурации.....	30
Подготовка.....	30
Как это делается.....	30
Как это работает.....	31
Дополнительно.....	31
См. также.....	33
Сохранение любого значения в контейнере или переменной.....	33
Подготовка.....	34
Как это делается.....	34
Как это работает.....	34
Дополнительно.....	35
См. также.....	36
Хранение одного из нескольких выбранных типов в контейнере или переменной.....	36
Подготовка.....	36
Как это делается.....	36
Как это работает.....	37
Дополнительно.....	37
См. также.....	38
Использование более безопасного способа работы с контейнером, в котором хранится один из нескольких выбранных типов.....	38
Подготовка.....	40
Как это делается.....	40
Как это работает.....	41
Дополнительно.....	41
См. также.....	42

Возврат значения или флага «значения нет»	42
Подготовка	42
Как это делается.....	42
Как это работает.....	43
Дополнительно.....	44
См. также	44
Возвращение массива из функции.....	44
Подготовка	44
Как это делается.....	44
Как это работает.....	45
Дополнительно.....	45
См. также	46
Объединение нескольких значений в одно	46
Подготовка	46
Как это делается.....	46
Как это работает.....	48
Дополнительно.....	48
См. также	48
Привязка и переупорядочение параметров функции	49
Подготовка	49
Как это делается.....	49
Как это работает.....	50
Дополнительно.....	51
См. также	52
Получение удобочитаемого имени типа	52
Подготовка	52
Как это делается.....	52
Как это работает.....	53
Дополнительно.....	53
См. также	54
Использование эмуляции перемещения C++11	54
Подготовка	54
Как это делается.....	54
Как это работает.....	56
Дополнительно.....	56
См. также	56
Создание не копируемого класса	57
Подготовка	57
Как это делается.....	57
Как это работает.....	58
См. также	58
Создание не копируемого, но перемещаемого класса	58
Подготовка	59
Как это делается.....	59
Как это работает.....	61
Дополнительно.....	61
См. также	62

Использование алгоритмов C++14 и C++11	62
Подготовка	62
Как это делается.....	63
Как это работает.....	63
Дополнительно.....	63
См. также	64
Глава 2. Управление ресурсами	65
Вступление	65
Управление указателями на классы, которые не покидают область видимости	65
Подготовка	66
Как это делается.....	66
Как это работает.....	67
Дополнительно.....	68
См. также	69
Подсчет указателей на классы	69
Подготовка	69
Как это делается.....	70
Как это работает.....	70
Дополнительно.....	71
См. также	72
Управление указателями на массивы, которые не покидают область видимости	72
Подготовка	72
Как это делается.....	73
Как это работает.....	73
Дополнительно.....	73
См. также	74
Подсчет указателей на массивы	74
Подготовка	75
Как это делается.....	75
Как это работает.....	77
Дополнительно.....	77
См. также	77
Хранение любых функциональных объектов в переменной	78
Подготовка	78
Как это делается.....	78
Как это работает.....	79
Дополнительно.....	79
См. также	80
Передача указателя на функцию	80
Подготовка	80
Как это делается.....	80
Как это работает.....	80
Дополнительно.....	80
См. также	81

Хранение любых лямбда-функций C++11 в переменной	81
Подготовка	81
Как это делается.....	81
Дополнительно.....	81
См.также.....	82
Контейнеры указателей	82
Подготовка	84
Как это делается.....	84
Как это работает.....	84
Дополнительно.....	84
См. также	85
Делайте это при выходе из области видимости!	86
Подготовка	86
Как это делается.....	86
Как это работает.....	87
Дополнительно.....	87
См. также	88
Инициализация базового класса членом класса-наследника	88
Подготовка	89
Как это делается.....	89
Как это работает.....	89
Дополнительно.....	90
См. также	90
Глава 3. Преобразование и приведение	91
Вступление	91
Преобразование строк в числа	91
Подготовка	92
Как это делается.....	92
Как это работает.....	92
Дополнительно.....	93
См. также	94
Преобразование чисел в строки	94
Подготовка	94
Как это делается.....	94
Как это работает.....	95
Дополнительно.....	95
См. также	96
Преобразование чисел в числа	96
Подготовка	97
Как это делается.....	97
Как это работает.....	98
Дополнительно.....	98
См. также	99
Преобразование пользовательских типов в строки и из строк.....	99
Как это делается.....	99
Дополнительно.....	101
См. также	101

Приведение умных указателей.....	102
Подготовка	102
Как это делается.....	102
Как это работает.....	103
Дополнительно... ..	103
См. также	103
Приведение полиморфных объектов	103
Подготовка	104
Как это делается.....	104
Как это работает.....	104
Дополнительно... ..	104
См. также	105
Синтаксический анализ (parsing) простого ввода.....	105
Подготовка	106
Как это делается.....	106
Как это работает.....	107
Дополнительно... ..	108
См. также	110
Синтаксический анализ (parsing) сложного ввода	110
Подготовка	110
Как это делается.....	110
Как это работает.... ..	113
Дополнительно... ..	114
См. также	114
Глава 4. Уловки времени компиляции	115
Вступление	115
Проверка размеров во время компиляции.....	115
Подготовка	116
Как это делается.....	116
Как это работает.....	116
Дополнительно... ..	118
См. также	119
Активация использования шаблона функции для интегральных типов	120
Подготовка	120
Как это делается.....	120
Как это работает.....	121
Дополнительно... ..	122
См. также	123
Отключение использования шаблона функции для действительных типов....	123
Подготовка	124
Как это делается.....	124
Как это работает.....	124
Дополнительно... ..	125
См. также	125
Создание типа из числа.....	125
Подготовка	126

Как это делается.....	126
Как это работает.....	127
Дополнительно.....	127
См. также	128
Реализация свойства типов (type trait)	128
Подготовка	128
Как это делается.....	128
Как это работает.....	128
Дополнительно.....	129
См. также	129
Выбор оптимального оператора для параметра шаблона.....	129
Подготовка	130
Как это делается.....	130
Как это работает.....	131
Дополнительно.....	131
См. также	133
Получение типа выражения в C++03	133
Подготовка	133
Как это делается.....	133
Как это работает.....	134
Дополнительно.....	134
См. также	135
Глава 5. Многопоточность.....	137
Вступление	137
Создание потока выполнения	137
Подготовка	138
Как это делается.....	138
Как это работает.....	138
Дополнительно.....	139
См. также	141
Синхронизация доступа к общему ресурсу	141
Подготовка	142
Как это делается.....	142
Как это работает.....	143
Дополнительно.....	144
См. также	145
Быстрый доступ к общему ресурсу с использованием атомарных операций	145
Подготовка	145
Как это делается.....	146
Как это работает.....	146
Дополнительно.....	147
См. также	148
Создание класса work_queue	148
Подготовка	148
Как это делается.....	148

Как это работает.....	150
Дополнительно.....	151
См. также	152
Блокировка «Несколько читателей – один писатель».....	152
Подготовка	153
Как это делается.....	154
Как это работает.....	154
Дополнительно.....	155
См. также	155
Создание переменных, уникальных для каждого потока	155
Подготовка	156
Как это делается.....	156
Как это работает.....	157
Дополнительно.....	157
См. также	157
Прерывание потока	157
Подготовка	158
Как это делается.....	158
Как это работает.....	158
Дополнительно.....	159
См. также	159
Манипулирование группой потоков	159
Подготовка	160
Как это делается.....	160
Как это работает.....	160
Дополнительно.....	160
См. также	161
Безопасная инициализация общей переменной	161
Подготовка	162
Как это делается.....	162
Как это работает.....	163
Дополнительно.....	163
См. также	164
Захват нескольких мьютексов	164
Подготовка	165
Как это делается.....	165
Как это работает.....	165
Дополнительно.....	166
См. также	166
Глава 6. Манипулирование задачами	167
Вступление	167
Прежде чем вы начнете.....	167
Регистрация задачи для обработки произвольного типа данных	168
Подготовка	168
Как это делается.....	168

Как это работает.....	170
Дополнительно.....	171
См. также	171
Создание таймеров и обработка событий таймера в качестве задач	172
Подготовка	172
Как это делается.....	172
Как это работает.....	174
Дополнительно.....	175
См. также	176
Передача данных по сети в качестве задачи	176
Подготовка	176
Как это делается.....	176
Как это работает.....	180
Дополнительно.....	183
См. также	184
Прием входящих соединений.....	184
Подготовка	184
Как это делается.....	184
Как это работает.....	186
Дополнительно.....	187
См. также	188
Параллельное выполнение различных задач.....	188
Приступаем.....	188
Как это делается.....	188
Как это работает.....	189
Дополнительно.....	190
См. также	190
Конвейерная обработка задач	190
Подготовка	191
Как это делается.....	191
Как это работает.....	193
Дополнительно.....	194
См. также	194
Создание неблокирующего барьера.....	194
Подготовка	196
Как это делается.....	196
Как это работает.....	197
Дополнительно.....	197
См. также	198
Хранение исключения и создание задачи из него.....	198
Подготовка	198
Как это делается.....	198
Как это работает.....	200
Дополнительно.....	200
См. также	201
Получение и обработка системных сигналов в качестве задач	201
Подготовка	202

Как это делается.....	202
Как это работает.....	203
Дополнительно.....	204
См. также	204
Глава 7. Манипулирование строками	205
Вступление	205
Смена регистра символов и сравнение без учета регистра.....	205
Подготовка	205
Как это делается.....	206
Как это работает.....	207
Дополнительно.....	207
См. также	207
Сопоставление строк с использованием регулярных выражений.....	207
Приступим	208
Как это делается.....	208
Как это работает.....	210
Дополнительно.....	210
См. также	211
Поиск и замена строк с использованием регулярных выражений.....	211
Подготовка	212
Как это делается.....	212
Как это работает.....	213
Дополнительно.....	213
См. также	213
Форматирование строк с использованием безопасных printf-подобных функций	214
Подготовка	214
Как это делается.....	214
Как это работает.....	215
Дополнительно.....	215
См. также	216
Замена и стирание строк.....	216
Подготовка	216
Как это делается.....	216
Как это работает.....	217
Дополнительно.....	217
См. также	218
Представление строки двумя итераторами.....	218
Подготовка	218
Как это делается.....	218
Как это работает.....	219
Дополнительно.....	220
См. также	220
Использование типа «ссылка на строку»	221
Подготовка	221
Как это делается.....	221

Как это работает.....	222
Дополнительно.....	224
См. также	224
Глава 8. Метапрограммирование	225
Вступление	225
Использование типа «вектор типов».....	225
Подготовка	226
Как это делается.....	226
Как это работает.....	228
Дополнительно.....	229
См. также	229
Манипулирование вектором типов.....	230
Подготовка	230
Как это делается.....	230
Как это работает.....	232
Дополнительно.....	233
См. также	234
Получение результирующего типа функции во время компиляции.....	234
Подготовка	235
Как это делается.....	235
Как это работает.....	236
Дополнительно.....	236
См. также	236
Создание метафункции высшего порядка	236
Подготовка	237
Как это делается.....	237
Как это работает.....	238
Дополнительно.....	238
См. также	239
Ленивое вычисление метафункции	239
Подготовка	239
Как это делается.....	239
Как это работает.....	240
Дополнительно.....	242
См. также	242
Преобразование всех элементов кортежа в строки	242
Подготовка	242
Как это делается.....	242
Как это работает.....	244
Дополнительно.....	244
См. также	246
Расщепление кортежей	246
Подготовка	246
Как это делается.....	246
Как это работает.....	247

Дополнительно.....	248
См. также	249
Манипулирование гетерогенными контейнерами в C++14	249
Подготовка	249
Как это делается.....	249
Как это работает.....	250
Дополнительно.....	251
См. также	252
Глава 9. Контейнеры.....	253
Вступление	253
Хранение нескольких элементов в контейнере	253
Подготовка	254
Как это делается.....	254
Как это работает.....	255
Дополнительно.....	255
См. также	256
Хранение не более N элементов в контейнере.....	256
Подготовка	256
Как это делается.....	256
Как это работает.....	256
Дополнительно.....	257
См. также	258
Сверхбыстрое сравнение строк	258
Подготовка	259
Как это делается.....	259
Как это работает.....	260
Дополнительно.....	261
См. также	262
Использование неупорядоченных ассоциативных контейнеров	262
Подготовка	262
Как это делается.....	262
Как это работает.....	263
Дополнительно.....	265
См. также	265
Создание ассоциативного контейнера с индексированием и по значениям.....	266
Подготовка	266
Как это делается.....	266
Как это работает.....	268
Дополнительно.....	269
См. также	269
Использование многоиндексных контейнеров	269
Подготовка	270
Как это делается.....	270
Как это работает.....	272
Дополнительно.....	274
См. также	274

Получение преимуществ от односвязного списка и пула памяти	274
Подготовка	274
Как это делается.....	274
Как это работает.....	276
Дополнительно.....	277
См. также	277
Использование плоских ассоциативных контейнеров	278
Подготовка	278
Как это делается.....	278
Как это работает.....	279
Дополнительно.....	280
См. также	281
Глава 10. Сбор информации о платформе и компиляторе	283
Вступление	283
Обнаружение ОС и компилятора.....	284
Подготовка	284
Как это делается.....	284
Как это работает.....	284
Дополнительно.....	285
См. также	285
Обнаружение поддержки 128-битных целых чисел.....	285
Подготовка	285
Как это делается.....	285
Как это работает.....	286
Дополнительно.....	286
См. также	287
Обнаружение и обход отключенной динамической идентификации типа данных	287
Подготовка	287
Как это делается.....	287
Как это работает.....	288
Дополнительно.....	289
См. также	289
Написание метафункций с использованием более простых методов.....	290
Подготовка	290
Как это делается.....	290
Как это работает.....	291
Дополнительно.....	291
См. также	291
Уменьшение размера кода и повышение производительности пользовательских типов в C++11	292
Подготовка	292
Как это делается.....	292
Как это работает.....	293
Дополнительно.....	293
См. также	293

Переносимый способ экспорта и импорта функций и классов	294
Подготовка	294
Как это делается.....	294
Как это работает.....	295
Дополнительно.....	296
См. также	296
Обнаружение версии Boost и получение новейших функций	297
Подготовка	297
Как это делается.....	297
Как это работает.....	298
Дополнительно.....	298
См. также	298
Глава 11. Работа с системой.....	299
Вступление	299
Перечисление файлов в каталоге	299
Подготовка	300
Как это делается.....	300
Как это работает.....	301
Дополнительно.....	301
См. также	302
Стирание и создание файлов и каталогов	302
Подготовка	302
Как это делается.....	302
Как это работает.....	303
Дополнительно.....	304
См. также	304
Написание и использование плагинов	304
Подготовка	304
Как это делается.....	304
Как это работает.....	305
Дополнительно.....	306
См. также	306
Получение backtrace – текущей последовательности вызовов	306
Приступим.....	307
Как это делается.....	307
Как это работает.....	308
Дополнительно.....	308
См. также	309
Быстрая передача данных из одного процесса в другой	309
Подготовка	309
Как это делается.....	309
Как это работает.....	310
Дополнительно.....	311
См. также	311
Синхронизация межпроцессного взаимодействия	312
Подготовка	312

Как это делается.....	312
Как это работает.....	314
Дополнительно.....	314
См. также	315
Использование указателей в общей памяти	315
Подготовка	315
Как это делается.....	315
Как это работает.....	316
Дополнительно.....	317
См. также	317
Самый быстрый способ чтения файлов.....	317
Подготовка	317
Как это делается.....	317
Как это работает.....	318
Дополнительно.....	319
См. также	319
Сопрограммы – сохранение состояния и откладывание выполнения	319
Подготовка	319
Как это делается.....	320
Как это работает.....	321
Дополнительно.....	322
См. также	323
Глава 12. Касаясь верхушки айсберга	325
Вступление	325
Работа с графами	326
Подготовка	326
Как это делается.....	326
Как это работает.....	327
Дополнительно.....	329
См. также	329
Визуализация графов	329
Подготовка	330
Как это делается.....	330
Как это работает.....	331
Дополнительно.....	331
См. также	332
Использование генератора истинно случайных чисел.....	332
Приступаем	332
Как это делается.....	332
Как это работает.....	333
Дополнительно.....	333
См. также	334
Использование переносных математических функций.....	334
Подготовка	334
Как это делается.....	334
Как это работает.....	334

Дополнительно.....	335
См. также	335
Написание тестовых случаев	335
Подготовка	336
Как это делается.....	336
Как это работает.....	336
Дополнительно.....	337
См. также	337
Объединение нескольких тестовых случаев в одном тестовом модуле	338
Подготовка	338
Как это делается.....	338
Как это работает.....	339
Дополнительно.....	339
См. также	339
Манипулирование изображениями	339
Подготовка	340
Как это делается.....	340
Как это работает.....	342
Дополнительно.....	343
См. также	343
Предметный указатель	344

Предисловие от издательства

Отзывы и пожелания

Мы всегда рады отзывам наших читателей. Расскажите нам, что вы думаете об этой книге – что понравилось или, может быть, не понравилось. Отзывы важны для нас, чтобы выпускать книги, которые будут для вас максимально полезны.

Вы можете написать отзыв прямо на нашем сайте www.dmkpress.com, зайдя на страницу книги, и оставить комментарий в разделе «Отзывы и рецензии». Также можно послать письмо главному редактору по адресу dmkpress@gmail.com, при этом напишите название книги в теме письма.

Если есть тема, в которой вы квалифицированы, и вы заинтересованы в написании новой книги, заполните форму на нашем сайте по адресу http://dmkpress.com/authors/publish_book/ или напишите в издательство по адресу dmkpress@gmail.com.

Список опечаток

Хотя мы приняли все возможные меры для того, чтобы удостовериться в качестве наших текстов, ошибки все равно случаются. Если вы найдете ошибку в одной из наших книг – возможно, ошибку в тексте или в коде, – мы будем очень благодарны, если вы сообщите нам о ней. Сделав это, вы избавите других читателей от расстройств и поможете нам улучшить последующие версии этой книги.

Если вы найдете какие-либо ошибки в коде, пожалуйста, сообщите о них главному редактору по адресу dmkpress@gmail.com, и мы исправим это в следующих тиражах.

Нарушение авторских прав

Пиратство в интернете по-прежнему остается насущной проблемой. Издательство «ДМК Пресс» очень серьезно относится к вопросам защиты авторских прав и лицензирования. Если вы столкнетесь в интернете с незаконно выполненной копией любой нашей книги, пожалуйста, сообщите нам адрес копии или веб-сайта, чтобы мы могли применить санкции.

Пожалуйста, свяжитесь с нами по адресу электронной почты dmkpress@gmail.com со ссылкой на подозрительные материалы.

Мы высоко ценим любую помощь по защите наших авторов, помогающую нам предоставлять вам качественные материалы.

Об авторе

Если вам интересно, кто такой Антон Полухин и можно ли доверять ему в вопросах обучения C++ и библиотекам Boost, то вот несколько фактов:

- Антон Полухин в настоящее время представляет Россию в международном комитете по стандартизации C++;
- он является автором нескольких библиотек Boost и поддерживает ряд старых библиотек Boost;
- он перфекционист: все исходные коды из книги проходят автоматическое тестирование на нескольких платформах с использованием различных стандартов C++.

Но давайте начнем с самого начала.

Антон Полухин родился в России. В детстве он мог говорить на русском и венгерском языках и изучал английский в школе. Со школьных лет участвовал в различных соревнованиях по математике, физике и химии и побеждал в них.

Дважды был принят в университет: один раз за участие в городской олимпиаде по математике и второй раз за то, что получил высокий балл по вступительным олимпиадам в вуз. В его университетской жизни был год, когда он вообще не участвовал в экзаменах: он получил «зачет автоматом» во всех дисциплинах, написав программы повышенной сложности по каждому предмету. Свою будущую жену он встретил в университете, который закончил с отличием.

Более трех лет работал в VoIP-компании, разрабатывая бизнес-логику для коммерческой альтернативы Asterisc. В то время он начал вносить свой вклад в Boost и стал сопровождающим библиотеки Boost.LexicalCast. Он также начал делать переводы на русский язык для Ubuntu Linux в то время.

Сегодня работает в компании Yandex.Taxi, помогает русскоговорящим людям с предложениями по стандартизации C++, продолжает вносить вклад в opensource-проекты и язык C++ в целом.

Его код можно найти в библиотеках Boost, таких как Any, Conversion, DLL, LexicalCast, Stacktrace, TypeTraits, Variant и др.

Счастлив в браке уже более семи лет.

Я хотел бы поблагодарить свою семью, особенно мою жену Ирину Полухину, за то, что она рисовала эскизы рисунков и диаграмм для этой книги.

Огромное спасибо Полу Энтони Бристоу за обзор первого издания данной книги и за то, что он прошел через безумное количество запятых, которые я использовал в первых черновиках.

Отдельное спасибо Глену Джозефу Фернандесу за то, что он предоставил много полезной информации и комментариев по второму изданию.

Что касается русского издания книги – неоценимую помощь оказал Кирилл Марков. За что ему отдельное спасибо!

Я также хотел бы поблагодарить всех членов сообщества Boost за написание этих замечательных библиотек и за то, что они открыли для меня удивительный мир C++.

О рецензентах

Глен Джозеф Фернандес работал инженером-программистом в компаниях Intel и Microsoft. Он является автором библиотеки *Boost.Align*, основным участником поддержки библиотек *Boost.SmartPointers* и *Boost.Core*, а также внес вклад в ряд других библиотек Boost. Участвует в поддержке стандарта C++, создавая документы по предложениям и отчеты о дефектах, и у него даже есть по крайней мере одна функция, принятая для будущего стандарта C++20 (P0674r1: расширение `make_shared` для поддержки массивов). Глен живет со своей женой Кэролайн и дочерью Айрин в США. Он окончил Университет Сиднея в Австралии, до этого жил в Новой Зеландии.

Марков Кирилл увлёкся программированием ещё в школе. Начал заниматься коммерческой разработкой ПО с ранних курсов университета. С тех пор освоил множество платформ, технологий и языков программирования, является full stack разработчиком, но предпочитает backend разработку. На данный момент живёт и трудится в Москве ведущим программистом в одной из крупнейших компаний. Проповедует педантичный формализованный подход к процессам разработки. Неисправимый любитель чая и интересной беседы.

Вступительное слово автора

Более 10 лет назад, когда я только начал осваивать C++, с хорошей литературой было очень тяжело. В итоге навыки C++ приходилось оттачивать, изучая исходные коды библиотеки Boost. Дело двигалось очень медленно, все было непонятно, документация и комментарии на английском не сильно помогали. С тех пор прошло уже много лет, библиотеки Boost отчасти стали стандартом C++ и продолжают развиваться, опережая по своим возможностям стандартную библиотеку C++ на десятки лет. Функционал внутри Boost огромен... и все так же непонятен для начинающих.

Эта книга содержит ответы на типичные вопросы:

- Как мне решить вот эту проблему?
- Как это работает?
- Как это устроено под капотом?
- Как бы поэкспериментировать, не заморачиваясь с настройкой окружения?
- А разве подобного нет в стандартной библиотеке?
- А какие есть хитрости при работе с этим инструментом?
- А есть ли способ решить это получше?
- А что еще почитать на эту тему?

Другими словами, эта книга – тот помощник, которого мне не хватало в свое время и которого не хватает многим разработчикам поныне.

Надеюсь, вам понравится!

*Антон Полухин,
представитель России в Международном комитете по стандартизации C++,
разработчик и автор многих библиотек Boost,
руководитель группы Общих Компонент в Яндекс.Такси,
сопредседатель PГ21 C++ и модератор <https://stdcpp.ru>,
спикер на конференциях PГ21, Corehard, C++ Russia,
корпоративный консультант по вопросам C++ <https://apolukhin.github.io/>,
автор этой книги :)*

Конец ознакомительного фрагмента.

Приобрести книгу можно

в интернет-магазине

«Электронный универс»

e-Univers.ru