

УДК 004.42
ББК 32.973
Б28

А

Батчер, Пол.

Б28 Семь моделей конкуренции и параллелизма за семь недель. Раскрываем тайны потоков / П. Батчер ; пер. с англ. А. Н. Киселева. — 2-е изд., эл. — 1 файл pdf : 362 с. — Москва : ДМК Пресс, 2023. — (The Pragmatic Programmers). — Систем. требования: Adobe Reader XI либо Adobe Digital Editions 4.5 ; экран 10". — Текст : электронный.

ISBN 978-5-89818-332-5

С появлением микропроцессоров, обладающих большим числом ядер, понимание конкуренции и параллелизма при разработке программного обеспечения стало еще более важным, чем прежде. В книге вы познакомитесь с преимуществами функционального программирования с точки зрения конкуренции, узнаете, как применять акторы для разработки распределенного программного обеспечения, и исследуете приемы параллельной обработки огромных объемов информации на нескольких процессорах. Эта книга поможет вам приобрести новые навыки в разработке программ, благодаря чему вы будете готовы решать сложные задачи в ближайшие несколько лет.

УДК 004.42
ББК 32.973

Электронное издание на основе печатного издания: Семь моделей конкуренции и параллелизма за семь недель. Раскрываем тайны потоков / П. Батчер ; пер. с англ. А. Н. Киселева. — Москва : ДМК Пресс, 2015. — 360 с. — (The Pragmatic Programmers). — ISBN 978-5-97060-244-7. — Текст : непосредственный.

Все права защищены. Любая часть этой книги не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения владельцев авторских прав.

Материал, изложенный в данной книге, многократно проверен. Но поскольку вероятность технических ошибок все равно существует, издательство не может гарантировать абсолютную точность и правильность приводимых сведений. В связи с этим издательство не несет ответственности за возможные ошибки, связанные с использованием книги.

В соответствии со ст. 1299 и 1301 ГК РФ при устранении ограничений, установленных техническими средствами защиты авторских прав, правообладатель вправе требовать от нарушителя возмещения убытков или выплаты компенсации.

ISBN 978-5-89818-332-5

© 2014 The Pragmatic Programmers, LLC.
© Оформление, перевод на русский язык
ДМК Пресс, 2015

А



ОГЛАВЛЕНИЕ

Положительные отзывы к книге «Семь моделей конкуренции и параллелизма за семь недель»	5
Предисловие	13
Благодарности	15
Вступление	17
О книге	17
Чем не является эта книга.....	18
Примеры кода	18
Примечание для пользователей IDE.....	19
Примечание для пользователей Windows.....	19
Ресурсы в Сети.....	19
Глава 1. Введение.....	21
Конкуренция или параллелизм?	21
Похожие, но разные.....	21
За рамками последовательного программирования.....	23
Параллельная архитектура	23
Параллелизм на уровне битов	24
Параллелизм на уровне инструкций	24
Параллелизм данных.....	24
Параллелизм на уровне задач	25
Конкуренция: за рамками множества ядер	26
Конкурентные программы для конкурентного мира	26
Распределенные программы для распределенного мира.....	27
Надежные программы для непредсказуемого мира.....	27
Простые программы в сложном мире	28
Семь моделей	28
Глава 2. Потоки выполнения и блокировки	31
Самое простое из того, что может работать.....	31
День 1: взаимoisключение и модели памяти.....	32
Создание потока	33
Наша первая блокировка	34
Загадочная память	37

Видимость памяти.....	38
Несколько блокировок.....	39
Опасности сторонних методов.....	43
В завершение первого дня.....	44
День 2: помимо встроенных блокировок.....	46
Прерываемое блокирование.....	47
Тайм-ауты.....	49
Блокирование методом перебора.....	51
Условные переменные.....	54
Атомарные переменные.....	57
В завершение второго дня.....	58
День 3: на плечах гигантов.....	60
Еще раз о создании потоков.....	61
Копирование при записи.....	62
Законченная программа.....	64
В завершение третьего дня.....	74
В завершение.....	75
Сильные стороны.....	75
Слабые стороны.....	76
Другие языки.....	78
Напоследок.....	79

Глава 3. Функциональное программирование..... 80

Если какие-то действия вредят вам, перестаньте выполнять их.....	80
День 1: программирование без изменяемого состояния.....	81
Опасности изменяемого состояния.....	81
Краткий экскурс в язык Clojure.....	84
Первая функциональная программа.....	86
Параллелизм без усилий.....	87
Функциональный подсчет слов.....	89
Лень – это благо.....	93
В завершение первого дня.....	94
День 2: функциональный параллелизм.....	95
По одной странице за раз.....	95
Разделение данных на пакеты для увеличения производительности.....	98
Редуценты (reducers).....	99
Внутреннее устройство редуцентов.....	100
Разделяй и властвуй.....	103
Поддержка функции fold.....	104
Подсчет слов с помощью fold.....	105
В завершение второго дня.....	107
День 3: функциональная конкуренция.....	108
Та же структура, разный порядок вычислений.....	108

Ссылочная прозрачность.....	109
Потоки данных.....	110
Механизм future.....	111
Механизм promise.....	112
Функциональная веб-служба.....	113
В завершение третьего дня.....	121
В завершение.....	122
Сильные стороны.....	124
Слабые стороны.....	124
Другие языки.....	124
Напоследок.....	125
Глава 4. Путь Clojure – разделение идентичности и состояния.....	126
Лучшее из двух миров.....	126
День 1: атомы и сохраненные структуры данных.....	127
Атомы.....	127
Многопоточная веб-служба с изменяемым состоянием.....	129
Сохраненные структуры данных.....	130
Идентичность или состояние.....	134
Повторения.....	134
Валидаторы.....	135
Функции-наблюдатели.....	135
Гибридная веб-служба.....	136
В завершение первого дня.....	140
День 2: агенты и программная транзакционная память.....	141
Агенты.....	141
Журнал в памяти.....	145
Программная транзакционная память.....	146
Изменяемое разделяемое состояние.....	151
В завершение второго дня.....	151
День 3: погружение в глубину.....	152
Решение задачи о философах на основе STM.....	153
Решение задачи о философах без применения STM.....	155
Атомы или STM?.....	157
Собственная реализация конкуренции.....	158
В завершение третьего дня.....	160
В завершение.....	161
Сильные стороны.....	161
Слабые стороны.....	162
Другие языки.....	162
Напоследок.....	162
Глава 5. Акторы.....	164
Не объекты, а скорее ориентированные на объекты.....	164

День 1: сообщения и почтовые ящики.....	166
Наш первый актер	166
Почтовые ящики и очереди	167
Прием сообщений	168
Связывание процессов	169
Актеры с сохранением состояния	170
Соккрытие сообщений за фасадом API.....	171
Двунаправленное взаимодействие.....	172
Именованние процессов	174
Отступление – функции первого порядка.....	176
Параллельная версия map().....	176
В завершение первого дня	177
День 2: обработка ошибок и отказоустойчивость.....	178
Актер кэширования	179
Определение момента отказа.....	182
Слежение за работой процессов	185
Тайм-ауты.....	186
Ядро ошибки	187
И пусть падает!.....	189
В завершение второго дня.....	190
День 3: распределенные приложения.....	191
ОТР	191
Узлы	196
Распределенный счетчик слов.....	200
В завершение третьего дня	206
В завершение.....	207
Сильные стороны	208
Слабые стороны	209
Другие языки	209
Напоследок	210

Глава 6. Взаимодействие последовательных процессов 211

Взаимодействия – это все.....	211
День 1: каналы и блоки go.....	213
Каналы	213
Блоки go	217
Операции с каналами	222
В завершение первого дня	225
День 2: множество каналов и ввод/вывод.....	227
Обслуживание множества каналов	227
Асинхронный опрос	230
Асинхронный ввод/вывод	233
В завершение второго дня.....	240
День 3: модель CSP на стороне клиента	241

Конкуренция – это образ жизни	242
Привет, ClojureScript	242
Обработка событий	245
Усмирение функций обратного вызова	247
Отправляемся в путь, чтобы увидеть Мастера	248
В завершение третьего дня	251
В завершение	251
Сильные стороны	252
Слабые стороны	252
Другие языки	253
Напоследок	253

Глава 7. Параллелизм данных 254

В недрах вашего ноутбука спрятан суперкомпьютер.....	254
День 1: программирование GPGPU.....	255
Обработка данных и параллелизм данных	255
Наша первая программа для OpenCL.....	258
Профилирование.....	264
Множество возвращаемых значений	265
Обработка ошибок.....	266
В завершение первого дня	268
День 2: многомерность и рабочие группы.....	270
Многомерные массивы рабочих элементов	270
Получение информации об устройстве	273
Модель платформы	275
Модель памяти.....	276
Параллельная свертка	277
Свертка с одной рабочей группой.....	277
В завершение второго дня.....	282
День 3: OpenCL и OpenGL – храните данные в GPU	283
Водная рябь	284
LWJGL.....	284
Отображение сетки в OpenGL	285
Доступ к буферу OpenGL из ядра OpenCL	287
Имитация ряби	288
В завершение третьего дня	291
В завершение.....	293
Сильные стороны	293
Слабые стороны	294
Другие языки	294
Напоследок	294

Глава 8. Лямбда-архитектура 295

Параллелизм позволяет обрабатывать гигантские объемы данных.....	295
---	-----

День 1: MapReduce	297
Практические аспекты	298
Основы Hadoop	299
Подсчет слов с помощью Hadoop	301
Опробование на Amazon EMR	305
Обработка XML	308
В завершение первого дня	310
День 2: пакетный уровень	313
Проблемы с традиционными системами данных	313
Вечные истины	315
Лучшие данные – исходные данные	315
Авторы правок в Википедии	317
Завершение картины	323
В завершение второго дня	325
День 3: уровень ускорения	327
Архитектура уровня ускорения	328
Подсчет правок с помощью Storm	335
В завершение третьего дня	341
В завершение	342
Сильные стороны	343
Слабые стороны	343
Альтернативы	343
Напоследок	343
Глава 9. В заключение	344
Куда мы идем?	344
Будущее за неизменяемостью	345
Будущее за распределенными вычислениями	346
Темы, оставшиеся за бортом	346
Fork/Join и захват задачи	347
Потоки данных	347
Реактивное программирование	347
Функциональное реактивное программирование	348
Grid-вычисления	348
Пространства кортежей	348
Выбор за вами	349
Библиография	350
Предметный указатель	352