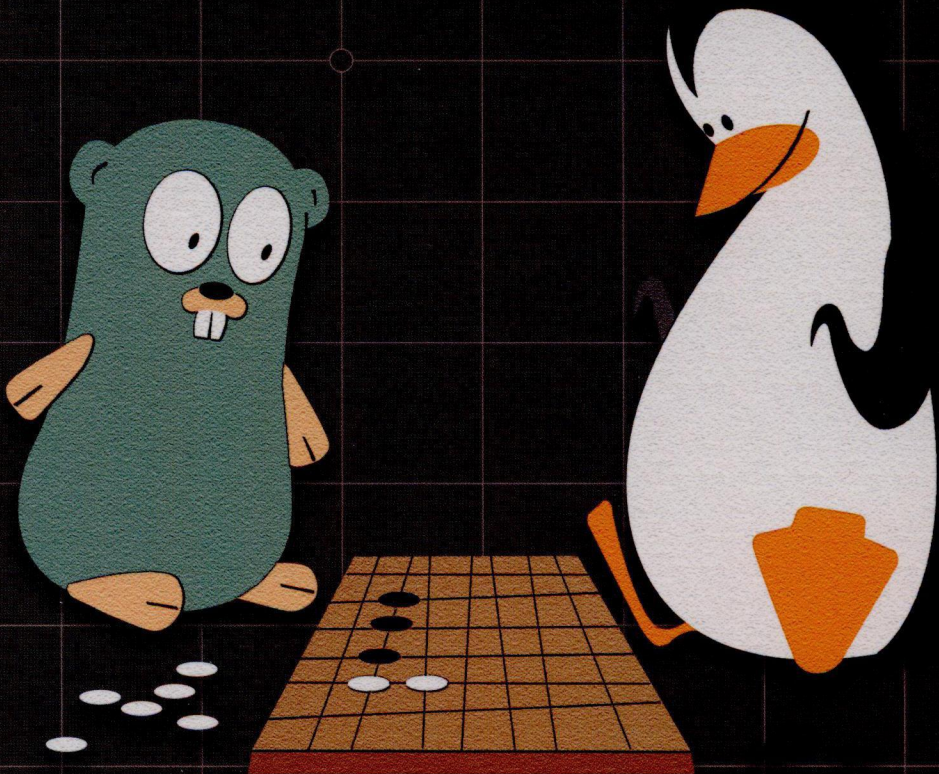


Олег Цилюрик

Linux и Go

эффективное низкоуровневое
программирование



Олег Цильюрик

Linux и Go

**эффективное низкоуровневое
программирование**

Санкт-Петербург

«БХВ-Петербург»

2023

УДК 004.43
ББК 32.973.26-018.2
Ц60

Цилюрик О. И.

Ц60 Linux и Go. Эффективное низкоуровневое программирование. — СПб.: БХВ-Петербург, 2023. — 304 с.: ил.

ISBN 978-5-9775-1741-6

Первая русскоязычная книга об интеграции языка Go в архитектуру ядра Linux. Содержит вводный курс по языку Go для системного программирования, описывает применение Go для реализации конкурентности и параллелизма, рассказывает об интероперабельности между имеющимся кодом на C с новым кодом на Go, а также исследует возможности внедрения Go для повышения производительности различных дистрибутивов.

Книга ориентирована на программистов и системных администраторов, работающих с Linux, будет интересна разработчикам ядра Linux и драйверов устройств.

УДК 004.43
ББК 32.973.26-018.2

Группа подготовки издания:

Руководитель проекта	<i>Олег Сивченко</i>
Зав. редакцией	<i>Людмила Гауль</i>
Редактор	<i>Юлия Попова</i>
Компьютерная верстка	<i>Натальи Смирновой</i>
Оформление обложки	<i>Зои Канторович</i>

Подписано в печать 06.04.23.
Формат 70×100^{1/16}. Печать офсетная. Усл. печ. л. 24,51.
Доп. тираж 700 экз. Заказ № 2590
"БХВ-Петербург", 191036, Санкт-Петербург, Гончарная ул., 20.

Отпечатано в АО "Первая Образцовая типография"
Филиал "Чеховский Печатный Двор"
142300, Московская область, г. Чехов, ул. Полиграфистов, д. 1
Сайт: www.chpd.ru, E-mail: sales@chpd.ru, тел. 8(499)270-73-59

ISBN 978-5-9775-1741-6

© ООО "БХВ", 2023
© Оформление. ООО "БХВ-Петербург", 2023

Оглавление

Предисловие	7
Предназначение и целевая аудитория	7
Код примеров и замеченные опечатки	9
Соглашения и выделения, принятые в тексте	10
Напоминание	11
Источники информации	11
 Часть I. Инструментарий языка Go	13
 Глава 1. Предыстория Go	15
«Отцы-основатели» о целях и мотивации	15
Применимость: беглый взгляд	16
Go, C, C++ и другие	17
Источники информации	22
 Глава 2. Разворачиваем экосистему Go	23
Создание среды	23
Стандартная инсталляция	23
Версии среды	25
Альтернативы	26
«Самая последняя» версия	27
Смена версий	30
Проверяем: простейшая программа	30
Проверяем на простейшем приложении	32
Библиотеки статические и динамические	34
Компиляция или интерпретация	34
Выбор: GoLang или GCC?	35
Инфраструктура GoLang	36
Команды go	36
Переменные окружения	39
Платформы, переносимость и кросс-компиляция	47

Стиль кодирования (автоформатирование — ffmt)	49
Сборка приложений (build)	51
Сценарии на языке Go (run)	51
Загрузка проектов из сети (get).....	52
Утилиты GoLang (tool)	57
Связь с кодом C (Cgo).....	60
Сторонний и дополнительный инструментарий	63
Источники информации.....	65
ГЛАВА 3. Неформально о синтаксисе Go	66
Типы данных.....	71
Переменные.....	73
Выполнение такого приложения	74
Повторные декларации и переприсваивания	76
Константы	77
Агрегаты данных	79
Массивы и срезы	79
Двухмерные массивы и срезы.....	84
Структуры	85
Таблицы (хэши)	87
Динамическое создание переменных	89
Конструкторы и составные литералы	90
Операции	91
Функции	94
Вариативные функции.....	98
Стек процедур завершения	100
Обобщенные функции	101
Функции высших порядков.....	102
Встроенные функции.....	104
Объектно-ориентированное программирование	106
Методы.....	107
Интерфейсы.....	113
Обработка ошибочных ситуаций	117
Структура пакетов (библиотек) Go.....	121
Функция init.....	125
Импорт для использования побочных эффектов	126
Некоторые полезные и интересные стандартные пакеты.....	126
Пакет runtime	127
Форматированный ввод-вывод.....	128
Строки и пакет strings.....	129
Большие числа.....	134
Автоматизированное тестирование.....	135
Источники информации.....	135

Часть II. Конкурентность и многопроцессорность	137
Глава 4. Процессоры в Linux	139
Процессоры, ядра и гипертрэдинг	142
Загадочная нумерация процессоров	144
Управление процессорами Linux	147
Привязка к процессору	147
Как происходит диспетчирование в Linux	150
Источники информации	156
Глава 5. Параллелизм и многопроцессорность	157
Эволюция модели параллелизма	157
Параллельные процессы и fork	157
Потоки ядра и pthread_t POSIX	160
Сопрограммы — модель параллелизма Go	166
Параллелизм в Go	167
Сопрограммы — как это выглядит	168
Возврат значений функцией	171
Ретроспектива: сопрограммы в C++	172
Каналы	172
Функциональные замыкания в сопрограммах	177
Примитивы синхронизации	179
Конкурентность и параллельность	185
Источники информации	190
Глава 6. Масштабирование	191
Планирование активности сопрограмм	191
Испытательный стенд	193
Микрокомпьютеры (Single-Board Computers)	193
Рабочие десктопы	196
Серверы промышленного класса	197
Масштабирование в реале	198
1-я попытка	199
2-й подход к снаряду	205
О числе потоков исполнения	213
Источники информации	217
Часть III. Некоторые примеры и сравнения	219
Глава 7. Осваиваемся в синтаксисе Go	221
Утилита echo	221
Итерационное вычисление вещественного корня	223
Вычисление числа π	225

Случайная последовательность и ее моменты.....	227
Обсчет параметров 2D выпуклых многоугольников	230
Тривиальный web-сервер.....	235
Источники информации.....	236
ГЛАВА 8. Структуры данных, типы и их методы.....	237
Массивы и срезы.....	237
Многомерные срезы и массивы	242
Функции с множественным возвратом.....	243
ГЛАВА 9. Элементы функционального программирования	246
Функциональные замыкания	246
Карринг.....	253
Рекурсия	255
Рекурсия с кэшированием.....	255
Чистые функции	257
Источники информации.....	257
ГЛАВА 10. Скоростные и другие сравнения языков.....	258
Алгоритмические задачи для сравнения :	258
Некоторые известные алгоритмы	259
Числа Фибоначчи.....	260
Пузырьковая сортировка.....	263
Ханойская башня	268
Решето Эратосфена.....	272
ГЛАВА 11. Многопроцессорные параллельные вычисления.....	276
Скорость активации параллельных ветвей	276
Гонки	281
Защита критических данных.....	285
Многопроцессорный брутфорс	289
Каналы в сопрограмах	295
Таймеры.....	296
Тикеры	298
Когда не нужно злоупотреблять многопроцессорностью	299
Источники информации.....	302
Заключение.....	303
Об авторе.....	304