

УДК 004.6
ББК 32.973.26
К48

К48 Клеменс Бен
 Язык С в XXI веке / пер. с англ. А. А. Слинкина. – М.: ДМК Пресс, 2018. – 376 с.: ил.

ISBN 978-5-97060-614-8

Язык С — не просто фундамент всех современных языков программирования, он и сам — современный язык, идеальный для написания эффективных приложений передового уровня. Последние 20 лет С не стоял на месте. Сам язык и окружающая его экосистема подвергаются пересмотру. Эта книга начинается там, где другие заканчиваются. В ней рассказано, как изменилась функциональность, поддерживаемая любым компилятором, благодаря двум новым стандартам С, вышедшим со времен оригинального ANSI. Цель книги — рассмотреть то, чего нет в других учебниках по С: инструменты и окружение; библиотеки для работы со связанными списками и анализаторами XML; написание удобочитаемого кода с дружественным программным интерфейсом.

Издание предназначено для программистов, имеющих опыт работы на каком-либо языке и обладающими базовыми знаниями о С.

УДК 004.6
ББК 32.973.26

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.

ISBN 978-1-491-90389-6 (анг.)
 ISBN 978-5-97060-614-8 (рус.)

Copyright © 2015 Ben Klemens
 © Оформление, перевод, ДМК Пресс, 2018

Содержание

Предисловие	11
Часть I ❖ Окружение	23
Глава 1 ❖ Настраиваем среду для компиляции	24
Работа с менеджером пакетов.....	25
Компиляция программ на С в Windows.....	27
POSIX в Windows.....	27
Компиляция программ на С при наличии подсистемы POSIX.....	28
Компиляция программ на С в отсутствие подсистемы POSIX	29
Как пройти в библиотеку?	30
Несколько моих любимых флагов.....	32
Пути.....	33
Компоновка во время выполнения	36
Работа с файлами makefile	36
Задание переменных	37
Правила.....	40
Сборка библиотек из исходного кода	43
Сборка библиотек из исходного кода (даже если системный администратор против)	45
Компиляция С-программы с помощью встроенного документа	46
Включение файлов-заголовков из командной строки	46
Универсальный заголовок	47
Встроенные документы	48
Компиляция из stdin	50
Глава 2 ❖ Отладка, тестирование, документирование	51
Работа с отладчиком	51
Отладка программы как детективная история.....	53
Переменные GDB.....	62
Распечатка структур.....	63
Использование Valgrind для поиска ошибок	67
Автономное тестирование.....	69
Использование программы в качестве библиотеки	72
Покрытие.....	73
Встроенная документация	74
Doxugen.....	74
Грамотное программирование с помощью CWEB.....	76
Проверка ошибок	78
Ошибки и пользователи.....	78
Учет контекста, в котором работает пользователь.....	80
Как следует возвращать уведомление об ошибке?	81

Глава 3 ❖ Создание пакета для проекта	83
Оболочка	84
Замена команд оболочки их выводом	84
Применение циклов <code>for</code> в оболочке для обработки набора файлов	86
Проверка наличия файла	88
Команда <code>fc</code>	90
Файлы <code>makefile</code> и скрипты оболочки	92
Создание пакета с помощью <code>Autotools</code>	95
Пример работы с <code>Autotools</code>	96
Описание <code>Makefile</code> с помощью <code>Makefile.am</code>	100
Скрипт <code>configure</code>	104
Глава 4 ❖ Управление версиями	108
Получение списка отличий с помощью <code>diff</code>	109
Объекты <code>Git</code>	110
Тайник	114
Деревья и их ветви	115
Объединение	116
Перемещение	117
Дистанционные репозитории	118
Глава 5 ❖ Мирное сосуществование	121
Динамическая загрузка	121
Ограничения динамической загрузки	124
Процесс	124
Писать так, чтобы можно было понять	124
Функция-обертка	125
Контрабанда структур данных через границу	126
Компоновка	128
Python как включающий язык	128
Компиляция и компоновка	129
Условный подкаталог для <code>Automake</code>	130
<code>Distutils</code> при поддержке <code>Autotools</code>	131
Часть II ❖ Язык	134
Глава 6 ❖ Ваш приятель – указатель	136
Автоматическая, статическая и динамическая память	136
Автоматическая	137
Статическая	137
Динамическая	137
Переменные для хранения постоянного состояния	140
Указатели без <code>malloc</code>	142

Структуры копируются, для массивов создаются псевдонимы.....	143
malloc и игры с памятью.....	146
Виноваты звезды.....	147
Все, что нужно знать об арифметике указателей	148
Typedef как педагогический инструмент.....	150

Глава 7 ❖ Несущественные особенности синтаксиса С, которым в учебниках уделяется чрезмерно много внимания 153

Ни к чему явно возвращать значение из main	154
Пусть объявления текут свободно	154
Меньше приведений	157
Перечисления и строки.....	159
Метки, goto, switch и break	160
К вопросу о goto.....	161
Предложение switch	163
Нерекомендуемый тип float	164
Сравнение чисел без знака.....	167
Безопасное преобразование строки в число	168

Глава 8 ❖ Важные особенности синтаксиса С, которые в учебниках часто не рассматриваются 171

Выращивание устойчивых и плодоносящих макросов	172
Приемы работы с препроцессором.....	176
Проверочные макросы	179
Защита заголовков.....	181
Компоновка с ключевыми словами static и extern.....	183
Переменные с внешней компоновкой в файлах-заголовках.....	184
Ключевое слово const	186
Форма существительное–прилагательное	187
Конфликты	187
Глубина.....	188
Проблема char const **	189

Глава 9 ❖ Текст 192

Безболезненная обработка строк с помощью asprintf.....	192
Безопасность	195
Константные строки	196
Расширение строк с помощью asprintf.....	197
Песнь о strtok.....	199
Unicode	203
Кодировка для программ на С	205
Библиотеки для работы с Unicode	206
Пример кода	208

Глава 10 ❖ Улучшенная структура	211
Составные литералы	212
Инициализация с помощью составных литералов.....	213
Макросы с переменным числом аргументов	213
Безопасное завершение списков.....	215
Несколько списков	216
ForEach	217
Векторизация функции	218
Позиционные инициализаторы	219
Инициализация массивов и структур нулями	221
Псевдонимы типов спешат на помощь	222
К вопросу о стиле	224
Возврат нескольких значений из функции.....	225
Извещение об ошибках	226
Гибкая передача аргументов функциям.....	228
Объявление своей функции по аналогии с printf.....	229
Необязательные и именованные аргументы.....	231
Доведение до ума бестолковой функции	233
Указатель на void и структура, на которую он указывает.....	239
Функции с обобщенными входными параметрами.....	239
Обобщенные структуры	244
Глава 11 ❖ Объектно-ориентированное программирование на C.....	249
Расширение структур и словарей.....	251
Реализация словаря	253
C без зазоров.....	257
Функции в структурах	261
V-таблицы	265
Область видимости	270
Закрытые элементы структуры	271
Перегрузка.....	272
_Generic.....	274
Подсчет ссылок.....	277
Пример: объект подстроки	277
Пример: основанная на агентах модель формирования групп	281
Заключение.....	288
Глава 12 ❖ Параллельные потоки	290
Окружение.....	291
Составные части.....	292
OpenMP	293
Компиляция для использования OpenMP.....	294

Интерференция	295
Map-reduce	296
Несколько задач	297
Поточная локальность	299
Локализация нестатических переменных	300
Разделяемые ресурсы	300
Атомы	305
Библиотека pthread	307
Атомы C	311
Атомарные структуры	315
Глава 13 ❖ Библиотеки	320
GLib	320
Стандарт POSIX	321
Разбор регулярных выражений	321
Использование mmap для очень больших наборов данных	326
Библиотека GNU Scientific Library	328
SQLite	331
Запросы	332
libxml и cURL	334
Эпилог	338
Приложение ❖ Основные сведения о языке C	339
Структура	339
В C необходим этап компиляции, состоящий из одной команды	340
Существует стандартная библиотека, это часть операционной системы	341
Существует препроцессор	341
Существуют комментарии двух видов	342
Нет ключевого слова print	342
Объявления переменных	342
Любая переменная должна быть объявлена	342
Даже функции необходимо объявлять или определять	343
Базовые типы можно агрегировать в массивы и структуры	344
Можно определять новые структурные типы	345
Можно узнать размер типа	346
Не существует специального типа строки	346
Функции и выражения	347
Правила видимости в C очень просты	347
Функция main имеет особый смысл	348
Большая часть работы программы на C сводится к вычислению выражений	348
При вычислении функций используются копии входных аргументов	349

Выражения заканчиваются точкой с запятой	349
Есть много сокращенных способов записи арифметических операций	349
В С понятие истины трактуется расширительно	350
Результатом деления двух целых всегда является целое	350
В С имеется тернарный условный оператор	351
Ветвления и циклы несильно отличаются от других языков	351
Цикл for – просто компактная форма цикла while	352
Указатели	353
Можно напрямую запросить блок памяти	354
Массивы – это просто блоки памяти, любой блок памяти можно использовать как массив	354
Указатель на скаляр – это по существу массив с одним элементом	355
Существует специальная нотация для доступа к полям структур по указателю	356
Указатели позволяют изменять аргументы функции	356
Любой объект где-то находится, и, значит, на него можно указать	357
Глоссарий	358
Библиография	363
Предметный указатель	365