

УДК 004.2  
ББК 32.972  
Б87

Б87 Рэндал Э. Брайант, Дэвид Р. О'Халларон

Компьютерные системы: архитектура и программирование. 3-е изд. / пер. с англ. А. Н. Киселева. – М.: ДМК Пресс, 2022. – 994 с.: ил.

**ISBN 978-5-97060-492-2**

В книге описываются стандартные элементы архитектуры, такие как центральный процессор, память, порты ввода-вывода, а также операционная система, компилятор, компоновщик и сетевое окружение. Демонстрируются способы представления данных и программ на машинном уровне, приемы оптимизации программ, особенности управления потоками выполнения и виртуальной памятью, а также методы сетевого и параллельного программирования. Приведенные в книге примеры для процессоров, совместимых с Intel (x86\_64), написаны на языке C и выполняются в операционной системе Linux.

Издание адресовано студентам и преподавателям по IT-специальностям, а также будет полезно разработчикам, желающим повысить свой профессиональный уровень и писать программы, эффективно использующие возможности компьютерной архитектуры.

Authorized translation from the English language edition, entitled *Computer Systems: A Programmer's Perspective*, 3rd Edition, by Randal E. Bryant and David R. O'Hallaron, published by Pearson Education, Inc, publishing as Pearson, Copyright © 2016.

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 978-0-13-409266-9 (англ.)

ISBN 978-5-97060-492-2 (рус.)

Copyright © 2016, 2011, and 2003 by  
Randal E. Bryant and David R. O'Hallaron, 2021  
© Оформление, перевод на русский язык,  
издание, ДМК Пресс, 2022

# Оглавление

<b>Предисловие от издательства .....</b>	<b>17</b>
<b>Вступление.....</b>	<b>18</b>
<b>Об авторах.....</b>	<b>34</b>
<b>Глава 1. Экскурс в компьютерные системы.....</b>	<b>36</b>
1.1. Информация – это биты + контекст.....	38
1.2. Программы, которые переводятся другими программами в различные формы .....	39
1.3. Как происходит компиляция .....	41
1.4. Процессоры читают и интерпретируют инструкции, хранящиеся в памяти .....	42
1.4.1. Аппаратная организация системы.....	42
1.4.2. Выполнение программы hello .....	44
1.5. Различные виды кеш-памяти .....	46
1.6. Устройства памяти образуют иерархию.....	47
1.7. Операционная система управляет работой аппаратных средств.....	48
1.7.1. Процессы .....	49
1.7.2. Потоки.....	50
1.7.3. Виртуальная память.....	51
1.7.4. Файлы.....	53
1.8. Обмен данными в сетях.....	53
1.9. Важные темы .....	55
1.9.1. Закон Амдала .....	56
1.9.2. Конкуренция и параллелизм .....	57
1.9.3. Важность абстракций в компьютерных системах.....	60
1.10. Итоги.....	61
Библиографические заметки.....	61
Решения упражнений .....	61
<b>Часть I</b>	
<b>Структура программы и ее выполнение .....</b>	<b>63</b>
<b>Глава 2. Представление информации и работа с ней.....</b>	<b>64</b>
2.1. Хранение информации.....	68
2.1.1. Шестнадцатеричная система счисления.....	68
2.1.2. Размеры данных .....	71
2.1.3. Адресация и порядок следования байтов .....	74
2.1.4. Представление строк .....	80
2.1.5. Представление программного кода .....	81
2.1.6. Введение в булеву алгебру .....	82
2.1.7. Битовые операции в С .....	85
2.1.8. Логические операции в С.....	87

2.1.9. Операции сдвига в С.....	88
<b>2.2. Целочисленные представления .....</b>	<b>90</b>
2.2.1. Целочисленные типы .....	91
2.2.2. Представление целых без знака .....	92
2.2.3. Представление в дополнительном коде.....	94
2.2.4. Преобразования между числами со знаком и без знака.....	99
2.2.5. Числа со знаком и без знака в С.....	104
2.2.6. Расширение битового представления числа .....	106
2.2.7. Усечение чисел .....	109
2.2.8. Советы по приемам работы с числами со знаком и без знака .....	111
<b>2.3. Целочисленная арифметика .....</b>	<b>113</b>
2.3.1. Сложение целых без знака .....	113
2.3.2. Сложение целых в дополнительном коде .....	118
2.3.3. Отрицание целых в дополнительном коде .....	123
2.3.4. Умножение целых без знака .....	124
2.3.5. Умножение целых в дополнительном коде .....	124
2.3.6. Умножение на константу.....	128
2.3.7. Деление на степень двойки .....	130
2.3.8. Заключительные размышления о целочисленной арифметике .....	134
<b>2.4. Числа с плавающей точкой.....</b>	<b>135</b>
2.4.1. Дробные двоичные числа.....	136
2.4.2. Представление значений с плавающей точкой в стандарте IEEE .....	139
2.4.3. Примеры чисел .....	141
2.4.4. Округление .....	146
2.4.5. Операции с плавающей точкой .....	148
2.4.6. Значения с плавающей точкой в С .....	150
<b>2.5. Итоги.....</b>	<b>151</b>
Библиографические заметки.....	152
Домашние задания.....	153
Правила представления целых чисел на битовом уровне .....	154
Правила представления чисел с плавающей точкой на битовом уровне .....	165
Решения упражнений .....	167
<b>Глава 3. Представление программ на машинном уровне .....</b>	<b>184</b>
3.1. Историческая перспектива .....	187
<b>3.2. Программный код.....</b>	<b>190</b>
3.2.1. Машинный код.....	190
3.2.2. Примеры кода .....	192
3.2.3. Замечание по форматированию.....	195
<b>3.3. Форматы данных.....</b>	<b>197</b>
<b>3.4. Доступ к информации .....</b>	<b>198</b>
3.4.1. Спецификаторы операндов .....	200
3.4.2. Инструкции перемещения данных .....	201
3.4.3. Примеры перемещения данных .....	205
3.4.4. Вталкивание данных в стек и выталкивание из стека .....	208
<b>3.5. Арифметические и логические операции.....</b>	<b>209</b>

3.5.1. Загрузка эффективного адреса .....	210
3.5.2. Унарные и бинарные операции .....	212
3.5.3. Операции сдвига .....	212
3.5.4. Обсуждение .....	213
3.5.5. Специальные арифметические операции .....	215
3.6. Управление .....	218
3.6.1. Флаги условий .....	218
3.6.2. Доступ к флагам .....	219
3.6.3. Инструкции перехода .....	222
3.6.4. Кодирование инструкций перехода .....	223
3.6.5. Реализация условного ветвления потока управления .....	225
3.6.6. Реализация условного ветвления потока данных .....	229
3.6.7. Циклы .....	235
3.6.8. Оператор switch .....	245
3.7. Процедуры .....	250
3.7.1. Стек времени выполнения .....	251
3.7.2. Передача управления .....	252
3.7.3. Передача данных .....	256
3.7.4. Локальные переменные на стеке .....	258
3.7.5. Локальные переменные в регистрах .....	260
3.7.6. Рекурсивные процедуры .....	262
3.8. Распределение памяти под массивы и доступ к массивам .....	264
3.8.1. Базовые принципы .....	264
3.8.2. Арифметика указателей .....	266
3.8.3. Вложенные массивы .....	267
3.8.4. Массивы фиксированных размеров .....	268
3.8.5. Массивы переменных размеров .....	271
3.9. Структуры разнородных данных .....	273
3.9.1. Структуры .....	273
3.9.2. Объединения .....	276
3.9.3. Выравнивание .....	279
3.10. Комбинирование инструкций управления потоком выполнения и передачи данных в машинном коде .....	282
3.10.1. Указатели .....	283
3.10.2. Жизнь в реальном мире: использование отладчика GDB .....	284
3.10.3. Ссылки на ячейки за границами выделенной памяти и переполнение буфера .....	286
3.10.4. Предотвращение атак методом переполнения буфера .....	290
3.10.5. Поддержка кадров стека переменного размера .....	295
3.11. Вычисления с плавающей точкой .....	298
3.11.1. Операции перемещения и преобразования данных .....	300
3.11.2. Операции с плавающей точкой в процедурах .....	305
3.11.3. Арифметические операции с плавающей точкой .....	305
3.11.4. Определение и использование констант с плавающей точкой .....	307
3.11.5. Поразрядные логические операции с числами с плавающей точкой .....	308
3.11.6. Операции сравнения значений с плавающей точкой .....	309

3.11.7. Заключительные замечания об операциях с плавающей точкой.....	312
3.12. Итоги .....	312
Библиографические заметки.....	313
Домашние задания.....	314
Решения упражнений .....	325
<b>Глава 4. Архитектура процессора .....</b>	<b>349</b>
4.1. Архитектура системы команд Y86-64.....	352
4.1.1. Состояние, видимое программисту .....	352
4.1.2. Инструкции Y86-64.....	353
4.1.3. Кодирование инструкций .....	355
4.1.4. Исключения в архитектуре Y86-64.....	360
4.1.5. Программы из инструкций Y86-64.....	361
4.1.6. Дополнительные сведения об инструкциях Y86-64.....	366
4.2. Логическое проектирование и язык HCL .....	368
4.2.1. Логические вентили .....	368
4.2.2. Комбинационные цепи и булевы выражения в HCL.....	369
4.2.3. Комбинационные цепи для слов и целочисленные выражения в HCL...	371
4.2.4. Принадлежность множеству .....	375
4.2.5. Память и синхронизация .....	375
4.3. Последовательные реализации Y86-64 (SEQ) .....	378
4.3.1. Организация обработки в несколько этапов .....	378
4.3.2. Аппаратная реализация последовательной архитектуры SEQ .....	387
4.3.3. Синхронизация в последовательной реализации SEQ .....	391
4.3.4. Реализация этапов в последовательной версии SEQ .....	394
4.4. Общие принципы конвейерной обработки .....	402
4.4.1. Вычислительные конвейеры.....	402
4.4.2. Подробное описание работы конвейера.....	404
4.4.3. Ограничения конвейерной обработки.....	406
4.4.4. Конвейерная обработка с обратной связью.....	408
4.5. Конвейерные реализации Y86-64 .....	409
4.5.1. SEQ+: переупорядочение этапов обработки.....	409
4.5.2. Добавление конвейерных регистров.....	411
4.5.3. Переупорядочение сигналов и изменение их маркировки.....	415
4.5.4. Прогнозирование следующего значения PC .....	416
4.5.5. Риски конвейерной обработки .....	418
4.5.6. Обработка исключений .....	431
4.5.7. Реализация этапов в PIPE .....	434
4.5.8. Управляющая логика конвейера.....	441
4.5.9. Анализ производительности .....	451
4.5.10. Незаконченная работа.....	454
4.6. Итоги.....	457
4.6.1. Имитаторы Y86-64.....	458
Библиографические заметки.....	458
Домашние задания.....	459
Решения упражнений .....	465

<b>Глава 5. Оптимизация производительности программ</b>	<b>478</b>
5.1. Возможности и ограничения оптимизирующих компиляторов	481
5.2. Выражение производительности программы	484
5.3. Пример программы	486
5.4. Устранение неэффективностей в циклах	490
5.5. Сокращение вызовов процедур	493
5.6. Устранение избыточных ссылок на память	495
5.7. Общее описание современных процессоров	498
5.7.1. Общие принципы функционирования	498
5.7.2. Производительность функционального блока	502
5.7.3. Абстрактная модель работы процессора	504
5.8. Развертывание циклов	510
5.9. Увеличение степени параллелизма	514
5.9.1. Несколько аккумуляторов	515
5.9.2. Переупорядочение операций	520
5.10. Обобщение результатов оптимизации комбинирующего кода	524
5.11. Некоторые ограничивающие факторы	525
5.11.1. Вытеснение регистров	525
5.11.2. Прогнозирование ветвлений и штрафы за ошибки предсказания	526
5.12. Понятие производительности памяти	530
5.12.1. Производительность операций загрузки	530
5.12.2. Производительность операций сохранения	531
5.13. Жизнь в реальном мире: методы повышения производительности	537
5.14. Выявление и устранение узких мест производительности	538
5.14.1. Профилирование программ	538
5.14.2. Использование профилировщика при выборе кода для оптимизации	540
5.15. Итоги	544
Библиографические заметки	545
Домашние задания	545
Решения упражнений	548
<b>Глава 6. Иерархия памяти</b>	<b>553</b>
6.1. Технологии хранения информации	554
6.1.1. Память с произвольным доступом	554
6.1.2. Диски	562
6.1.3. Твердотельные диски	572
6.1.4. Тенденции развития технологий хранения	574
6.2. Локальность	577
6.2.1. Локальность обращений к данным программы	577
6.2.2. Локальность выборки инструкций	579
6.2.3. В заключение о локальности	579
6.3. Иерархия памяти	581
6.3.1. Кеширование в иерархии памяти	582
6.3.2. В заключение об иерархии памяти	585
6.4. Кеш-память	586

6.4.1. Обобщенная организация кеш-памяти .....	586
6.4.2. Кеш с прямым отображением.....	588
6.4.3. Ассоциативные кеши.....	595
6.4.4. Полностью ассоциативные кеши.....	597
6.4.5. Проблемы с операциями записи .....	600
6.4.6. Устройство реальной иерархии кешей.....	601
6.4.7. Влияние параметров кеша на производительность .....	602
6.5. Разработка программ, эффективно использующих кеш .....	603
6.6. Все вместе: влияние кеша на производительность программ .....	608
6.6.1. Гора памяти.....	608
6.6.2. Переупорядочение циклов для улучшения пространственной локальности .....	612
6.6.3. Использование локальности в программах.....	615
6.7. Итоги .....	616
Библиографические заметки.....	616
Домашние задания.....	617
Решения упражнений .....	627
<b>Часть II</b>	
<b>Выполнение программ в системе .....</b>	<b>633</b>
<b>Глава 7. Связывание .....</b>	<b>634</b>
7.1. Драйверы компиляторов .....	636
7.2. Статическое связывание.....	637
7.3. Объектные файлы .....	638
7.4. Перемещаемые объектные файлы .....	638
7.5. Идентификаторы и таблицы имен.....	640
7.6. Разрешение ссылок .....	643
7.6.1. Как компоновщик разрешает ссылки на повторяющиеся имена .....	644
7.6.2. Связывание со статическими библиотеками.....	648
7.6.3. Как компоновщики разрешают ссылки на статические библиотеки.....	651
7.7. Перемещение .....	652
7.7.1. Записи перемещения .....	653
7.7.2. Перемещение ссылок.....	654
7.8. Выполняемые объектные файлы .....	657
7.9. Загрузка выполняемых объектных файлов.....	659
7.10. Динамическое связывание с разделяемыми библиотеками.....	660
7.11. Загрузка и связывание с разделяемыми библиотеками из приложений .....	662
7.12. Перемещаемый программный код .....	665
7.13. Подмена библиотечных функций .....	668
7.13.1. Подмена во время компиляции .....	669
7.13.2. Подмена во время компоновки.....	670
7.13.3. Подмена во время выполнения.....	671
7.14. Инструменты управления объектными файлами.....	673
7.15. Итоги .....	673

Библиографические заметки.....	674
Домашние задания.....	674
Решения упражнений .....	677
<b>Глава 8. Управление исключениями .....</b>	<b>680</b>
8.1. Исключения.....	682
8.1.1. Обработка исключений .....	683
8.1.2. Классы исключений.....	685
8.1.3. Исключения в системах Linux/x86-64 .....	687
8.2. Процессы .....	690
8.2.1. Логический поток управления .....	691
8.2.2. Конкурентные потоки управления.....	692
8.2.3. Изолированное адресное пространство .....	693
8.2.4. Пользовательский и привилегированный режимы .....	693
8.2.5. Переключение контекста .....	694
8.3. Системные вызовы и обработка ошибок .....	695
8.4. Управление процессами .....	696
8.4.1. Получение идентификатора процесса .....	697
8.4.2. Создание и завершение процессов .....	697
8.4.3. Утилизация дочерних процессов.....	701
8.4.4. Приостановка процессов.....	706
8.4.5. Загрузка и запуск программ .....	707
8.4.6. Запуск программ с помощью функций fork и execve .....	709
8.5. Сигналы .....	712
8.5.1. Терминология сигналов .....	714
8.5.2. Посылка сигналов .....	715
8.5.3. Получение сигналов .....	717
8.5.4. Блокировка и разблокировка сигналов.....	720
8.5.5. Обработка сигналов.....	721
8.5.6. Синхронизация потоков во избежание неприятных ошибок конкурентного выполнения .....	730
8.5.7. Явное ожидание сигналов .....	732
8.6. Нелокальные переходы .....	735
8.7. Инструменты управления процессами.....	739
8.8. Итоги.....	739
Библиографические заметки.....	740
Домашние задания.....	740
Решения упражнений .....	747
<b>Глава 9. Виртуальная память .....</b>	<b>750</b>
9.1. Физическая и виртуальная адресация .....	752
9.2. Пространства адресов.....	753
9.3. Виртуальная память как средство кеширования.....	754
9.3.1. Организация кеша DRAM.....	754
9.3.2. Таблицы страниц .....	755
9.3.3. Попадание в кеш DRAM .....	756
9.3.4. Промах кеша DRAM .....	757



9.3.5. Размещение страниц.....	758
9.3.6. И снова о локальности.....	759
9.4. Виртуальная память как средство управления памятью.....	759
9.5. Виртуальная память как средство защиты памяти.....	761
9.6. Преобразование адресов.....	762
9.6.1. Интегрирование кешей и виртуальной памяти.....	765
9.6.2. Ускорение трансляции адресов с помощью TLB.....	766
9.6.3. Многоуровневые таблицы страниц.....	767
9.6.4. Все вместе: сквозное преобразование адресов.....	769
9.7. Практический пример: система памяти Intel Core i7/Linux.....	773
9.7.1. Преобразование адресов в Core i7.....	774
9.7.2. Система виртуальной памяти Linux.....	776
9.8. Отображение в память.....	780
9.8.1. И снова о разделяемых объектах.....	781
9.8.2. И снова о функции fork.....	783
9.8.3. И снова о функции execve.....	783
9.8.4. Отображение в память на уровне пользователя с помощью функции mmap.....	785
9.9. Динамическое распределение памяти.....	786
9.9.1. Функции malloc и free.....	787
9.9.2. Что дает динамическое распределение памяти.....	790
9.9.3. Цели механизмов распределения памяти и требования к ним.....	791
9.9.4. Фрагментация.....	792
9.9.5. Вопросы реализации.....	793
9.9.6. Неявные списки свободных блоков.....	794
9.9.7. Размещение распределенных блоков.....	796
9.9.8. Разбиение свободных блоков.....	796
9.9.9. Увеличение объема динамической памяти.....	797
9.9.10. Объединение свободных блоков.....	797
9.9.11. Объединение с использованием граничных тегов.....	798
9.9.12. Все вместе: реализация простого механизма распределения памяти.....	800
9.9.13. Явные списки свободных блоков.....	807
9.9.14. Раздельные списки свободных блоков.....	808
9.10. Сборка мусора.....	811
9.10.1. Основы сборки мусора.....	811
9.10.2. Алгоритм сборки мусора Mark&Sweep.....	813
9.10.3. Консервативный алгоритм Mark&Sweep для программ на C.....	814
9.11. Часто встречающиеся ошибки.....	815
9.11.1. Разыменование недопустимых указателей.....	815
9.11.2. Чтение неинициализированной области памяти.....	816
9.11.3. Переполнение буфера на стеке.....	816
9.11.4. Предположение о равенстве размеров указателей и объектов, на которые они указывают.....	816
9.11.5. Ошибки занижения или завышения на единицу.....	817
9.11.6. Ссылка на указатель вместо объекта.....	817

9.11.7. Неправильное понимание арифметики указателей .....	818
9.11.8. Ссылки на несуществующие переменные .....	818
9.11.9. Ссылка на данные в свободных блоках .....	818
9.11.10. Утечки памяти .....	819
9.12. Итоги .....	819
Библиографические заметки .....	820
Домашние задания .....	821
Решения упражнений .....	824

### Часть III

<b>Взаимодействие программ .....</b>	<b>829</b>
--------------------------------------	------------

<b>Глава 10. Системный уровень ввода/вывода .....</b>	<b>830</b>
-------------------------------------------------------	------------

10.1. Ввод/вывод в Unix .....	831
10.2. Файлы .....	832
10.3. Открытие и закрытие файлов .....	833
10.4. Чтение и запись файлов .....	835
10.5. Надежные чтение и запись с помощью пакета RIO .....	836
10.5.1. Функции RIO небуферизованного ввода/вывода .....	837
10.5.2. Функции RIO буферизованного ввода .....	838
10.6. Чтение метаданных файла .....	842
10.7. Чтение содержимого каталога .....	843
10.8. Совместное использование файлов .....	845
10.9. Переадресация ввода/вывода .....	848
10.10. Стандартный ввод/вывод .....	849
10.11. Все вместе: какие функции ввода/вывода использовать? .....	849
10.12. Итоги .....	851
Библиографические заметки .....	852
Домашние задания .....	852
Решения упражнений .....	853

<b>Глава 11. Сетевое программирование .....</b>	<b>854</b>
-------------------------------------------------	------------

11.1. Программная модель клиент-сервер .....	854
11.2. Компьютерные сети .....	855
11.3. Всемирная сеть интернет .....	860
11.3.1. IP-адреса .....	861
11.3.2. Доменные имена интернета .....	863
11.3.3. Интернет-соединения .....	866
11.4. Интерфейс сокетов .....	867
11.4.1. Структуры адресов сокетов .....	868
11.4.2. Функция socket .....	869
11.4.3. Функция connect .....	869
11.4.4. Функция bind .....	870
11.4.5. Функция listen .....	870
11.4.6. Функция accept .....	870
11.4.7. Преобразование имен хостов и служб .....	872
11.4.8. Вспомогательные функции для интерфейса сокетов .....	876

11.4.9. Примеры эхо-клиента и эхо-сервера .....	879
11.5. Веб-серверы.....	881
11.5.1. Основные сведения о вебе .....	881
11.5.2. Веб-контент.....	882
11.5.3. Транзакции HTTP.....	884
11.5.4. Обслуживание динамического контента .....	886
11.6. Все вместе: разработка небольшого веб-сервера TINY .....	889
11.7. Итоги .....	896
Библиографические заметки.....	896
Домашние задания.....	897
Решения упражнений .....	898
<b>Глава 12. Конкурентное программирование.....</b>	<b>901</b>
12.1. Конкурентное программирование с процессами.....	903
12.1.1. Конкурентный сервер, основанный на процессах .....	904
12.1.2. Достоинства и недостатки подхода на основе процессов .....	905
12.2. Конкурентное программирование с мультиплексированием ввода/вывода .....	906
12.2.1. Конкурентный на основе мультиплексирования ввода/вывода, управляемый событиями .....	909
12.2.2. Достоинства и недостатки мультиплексирования ввода/вывода .....	913
12.3. Конкурентное программирование с потоками выполнения.....	914
12.3.1. Модель выполнения многопоточных программ .....	914
12.3.2. Потоки Posix.....	915
12.3.3. Создание потоков .....	916
12.3.4. Завершение потоков.....	916
12.3.5. Утилизация завершившихся потоков .....	917
12.3.6. Обособление потоков .....	917
12.3.7. Инициализация потоков .....	918
12.3.8. Конкурентный многопоточный сервер.....	918
12.4. Совместное использование переменных несколькими потоками выполнения .....	920
12.4.1. Модель памяти потоков .....	921
12.4.2. Особенности хранения переменных в памяти.....	921
12.4.3. Совместно используемые переменные.....	922
12.5. Синхронизация потоков выполнения с помощью семафоров .....	922
12.5.1. Граф выполнения.....	925
12.5.2. Семафоры .....	928
12.5.3. Использование семафоров для исключительного доступа к ресурсам.....	929
12.5.4. Использование семафоров для организации совместного доступа к ресурсам.....	930
12.5.5. Все вместе: конкурентный сервер на базе предварительно созданных потоков.....	935
12.6. Использование потоков выполнения для организации параллельной обработки .....	938
12.7. Другие вопросы конкурентного выполнения .....	944

12.7.1. Безопасность в многопоточном окружении .....	944
12.7.2. Реентерабельность .....	946
12.7.3. Использование библиотечных функций в многопоточных программах.....	947
12.7.4. Состояние гонки.....	948
12.7.5. Взаимоблокировка (тупиковые ситуации).....	950
12.8. Итоги.....	953
Библиографические заметки.....	953
Домашние задания.....	954
Решения упражнений .....	958
<b>Приложение А. Обработка ошибок .....</b>	<b>963</b>
А.1. Обработка ошибок в системе Unix .....	963
А.2. Функции-обертки обработки ошибок.....	965
<b>Библиография.....</b>	<b>968</b>
<b>Предметный указатель .....</b>	<b>975</b>