

Оглавление

Предисловие	17
Почему нужно столько времени, чтобы поумнеть?	19
Проблемы, характерные для плохих программистов	20
Топ-7 отговорок плохих программистов	21
Несколько лет спустя	22
Что же вас ждет на следующих 393 страницах	23

Часть I. Привет, миф! Привет, мир!

Глава 1. Туда ли я попал?	26
Глава 2. От гордыни к смирению	29
Слабые стороны могут быть сильными	31
Истина не всегда кроется в сложном	33

Часть II. Программировать и понимать

Глава 3. Ты такой же, как все	36
Глава 4. Договоренности	38
Учить ли английский	39
Камень преткновения	42
Договоренности в команде	45
Глава 5. Присвоение имен	47
Правила присвоения имен	47
Сначала Византий, потом Константинополь, теперь Стамбул	49
Что должны уметь имена	51

Читать, понимать и не путать	52
Ясность и логика	54
Никаких шуток! Никакой крутости!	56
Материал, из которого делаются имена	57
Процессы, функции, методы	58
Переменные типа Boolean	67
Объектно-ориентированное программирование	69
Базы данных	70
Что же дальше?	72
Глава 6. Комментарии	74
Меньше слов — больше дела	76
О структуре комментариев	77
Комментарии к документации	79
Когда и что нужно комментировать	80
Случаи, когда комментарий мог бы помочь	82
Проблемные комментарии	86
Глава 7. Чтение кода	89
Думаете, это действительно того стоит?	89
Сперва читайте документацию	91
Распечатайте исходник	92
Схематично изобразите для себя, для чего предназначены некоторые компоненты программы	93
Сверху вниз, от простого к сложному	93
Учитесь искусству следопыта	94
Соотношение 80/20 оптимально (в большинстве случаев)	95
Помните о форматах данных	95
Истина в самой программе	96
Коллективное чтение кода	96
Глава 8. Где искать помощь	97
Подходящий момент	98
Спрашивайте в подходящем месте	100
Структурируйте запрос правильно	101

Думайте о читателе	104
Не ожидайте слишком многого.	105
Не приводите абстрактных примеров.	105
Оставайтесь вежливы несмотря ни на что	106
Глава 9. Право на оказание помощи.	108
Неправильная причина	108
Корыстные мотивы	110
Отсутствие эмпатии	111
Слишком много информации сразу.	112
Отвечайте на конкретные вопросы.	113
Если вы сами не знаете ответа.	114
Если ваши коллеги худшие программисты, чем вы	115
Плохой код? Сохраняйте спокойствие	116
Глава 10. Как выжить в команде.	118
Это не я!	120
Фактор автобуса	121
Сотрудничество с заказчиками.	123
Работа с добровольцами	123
Часть III. Работа над ошибками	
Глава 11. Искусство ошибаться для начинающих	126
В гостях у заблуждения: почувствуйте себя как дома.	127
Изучение ошибок в повседневности.	128
Базу данных съела собака!.	129
Шлем с мягкой подкладкой	130
Глава 12. Отладка I: поиск ошибок как наука	134
Систематический поиск ошибок	136
Наблюдение	137
Что усложняет наблюдение	139

Анализ и построение гипотез	140
Что усложняет построение гипотез	141
Проверка гипотез	142
Что усложняет проверку гипотез	143
Глава 13. Отладка II: найди ошибку.	145
Сообщения об ошибках — наши друзья	145
Кому тут что-то от меня нужно?	146
Инструменты и стратегии диагностики	148
Привычные подозреваемые	149
Валидаторы, Lint-программы, анализ кода	149
Отладка с помощью команды print.	150
Ведение журналов.	151
Отладчики.	153
Отладка в декларативных языках программирования.	159
Если больше ничего не помогает	162
Если и это не помогает	164
После поиска ошибок о поиске ошибок	164
Наиболее распространенные причины ошибок плохих программистов	165
Глава 14. Недобрые предзнаменования, или Коричневые M&M's	167
Слишком большие файлы	168
Очень длинные функции	169
Слишком широкие функции	169
Многоуровневые функции if/then	170
Всплывающие посреди кода числа	172
Сложные арифметические выражения в коде.	172
Глобальные переменные	173
Ремонтный код	174
Собственная имплементация функций	175
Особые случаи.	176
Неунифицированное написание	176

Функции более чем с пятью параметрами	177
Дублирование кода	178
Сомнительные названия файлов	178
Лабиринт чтения	178
Бесполезные комментарии	179
Очень большое количество базовых классов или интерфейсов	180
Очень большое количество переменных или свойств объекта	180
Блоки кода и функции, помещенные в комментарии	181
Использование определенного браузера	181
Подозрительные звуки клавиатуры	182
Глава 15. Рефакторинг	184
Писать ли заново?	185
Когда нужен рефакторинг	186
Все по порядку	189
Распределите код по нескольким файлам	194
Разбейте программный модуль на более мелкие модули	194
Устраняйте побочные эффекты	196
Объединяйте код	197
Понятнее оформляйте условия	201
Правильный цикл для правильной цели	203
Оформляйте циклы понятнее	204
Посмотрите критически на переменные	206
Рефакторинг баз данных	207
Что можно сделать еще	209
Стало ли теперь действительно лучше?	210
Когда от рефакторинга лучше отказаться	211
Проблема и ее решение	214
Глава 16. Тестирование	216
Для чего нужно тестирование	216
Способы тестирования	217
Системное тестирование	217

Модульное тестирование	219
Валидация данных	222
Тестирование производительности	224
Правильно тестировать	227
Глава 17. Предупреждающие знаки	228
GET- и POST-запросы	228
Кодирование знаков	230
Указание времени и даты.	231
Сохранение десятичных дробей в виде строковых, целочисленных и десятичных переменных	232
Передача переменных в виде значений или ссылки	234
Тяжелая работа с ничем	236
Рекурсия	237
Юзабилити	238
Глава 18. Компромиссы	240
Обманчивые достоинства	242
Уверенность в завтрашнем дне	242
Скорость	243
Совершенство, красота, элегантность	245
Абсолютизация: когда допустимы порочные практики	246

Часть IV. Выбор средств

Глава 19. Не делай сам	252
Что делать.	254
Библиотеки	255
Обращение с чужим кодом.	257
Чего не нужно делать самостоятельно	258
Чего ни в коем случае не следует делать самостоятельно	262
Три решения одной проблемы.	266

Глава 20. Инструментарий.	268
Редакторы	269
Какой язык программирования правильный?	270
REPL	274
Diff и patch	276
Менеджер пакетов.	279
Фреймворки.	281
Выбор фреймворка	282
Предупредительные знаки при работе с фреймворками	282
Среды разработки	284
Управление проектами	284
Проверка кода	285
Автодополнение кода	286
Помощь в организации труда.	288
Поиск и рефакторинг.	289
Недостатки	290
Глава 21. Система контроля версий	292
Альтернативы	294
Работа в VCS	295
Разрешение конфликтов	297
Какую систему контроля версий выбрать?	298
Subversion	299
Git.	299
Полезные идеи при работе с системой контроля версий.	300
Неудачные идеи при работе с системой контроля версий.	301
Система контроля версий — элемент ПО	302
Глава 22. Command and Conquer: из жизни командной строки	304
Повышение эффективности с помощью автоматизации	305
Наши длиннобородые предки	307
Windows	308
Что следует знать каждому программисту	308

Параметры	309
Аргументы.	310
Что делать с результатами?	311
Сопряжение результатов	311
Символы-джокеры	312
Навигация	313
Файлы	313
Просмотр.	315
Искать и найти	317
Сбережение ресурсов	319
Совместная работа	321
Синхронизация	321
Редактирование с сервера	322
Интернет	323
Стоит ли помнить обо всем этом?	324
Еще не всё!	325
Глава 23. Объектно-ориентированное программирование	326
Преимущества объектно-ориентированного программирования	328
Принципы объектно-ориентированного программирования	330
Модульность и инкапсуляция.	330
Абстракция данных	332
Полиморфизм	335
Наследование	335
Разумное использование ООП	337
Недостатки и проблемы	340
Различные модели объекта в зависимости от языка	341
Объектно-ориентированное программирование и планы по захвату мира.	341
Глава 24. Хранение данных.	343
Файлы	344
CSV/TSV	346
XML.	347

JSON	347
YAML	348
Системы управления версиями	349
Базы данных	349
Искать и найти	349
Реляционные базы данных	350
Базы данных NoSQL	353
Документоориентированные базы данных, такие как CouchDB или MongoDB	354
Графовые базы данных, такие как Neo4j	355
Глава 25. Безопасность	356
Важные концепции	357
Преимущества и недостатки открытости	359
Работа с паролями	361
Методы аутентификации	362
Внедрение SQL-кода и XSS — угрозы в пользовательском контенте	366
Белые списки лучше черных	371
Установите все ползунки на минимум	372
Черный ход тоже запирайте	374
Тестирование на проникновение	375
Ошибки других	376
Безопасность — это процесс	377
Глава 26. Полезные концепции	379
Исключения	379
Обработка ошибок	382
Состояние и отсутствие сохранения состояния	386
ID, GUID, UUID	387
Языковые семьи	389
Типы переменных	391
Разделение содержания и представления	394
Разделение на сервер разработки и «продакшн»-сервер	395
Селекторы	396

Пространства имен	398
Область видимости переменных.	400
Утверждения	401
Транзакции и откаты	404
Хеш-коды, дайджесты, цифровые отпечатки	404
CRUD и REST	407
Глава 27. Что дальше?	409
Кто такой хороший программист	410
Что еще можно почитать	411
Благодарности	413
Указатель	414