

Краткое содержание

Предисловие ко второму изданию	14
Предисловие	15
ЧАСТЬ I. СОВРЕМЕННОЕ СОСТОЯНИЕ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ	19
ГЛАВА 1. КОНЦЕПТУАЛЬНАЯ СХЕМА ЯЗЫКА ПРОГРАММИРОВАНИЯ	21
ГЛАВА 2. ПРИМЕР СОВРЕМЕННОГО БАЗОВОГО ЯП (МОДЕЛЬ А)	43
ГЛАВА 3. ВАЖНЕЙШИЕ АБСТРАКЦИИ: ДАННЫЕ, ОПЕРАЦИИ, СВЯЗЫВАНИЕ	69
ГЛАВА 4. ДАННЫЕ И ТИПЫ	85
ГЛАВА 5. РАЗДЕЛЬНАЯ КОМПИЛЯЦИЯ	145
ГЛАВА 6. АСИНХРОННЫЕ ПРОЦЕССЫ	151
ГЛАВА 7. НОТАЦИЯ	175
ГЛАВА 8. ИСКЛЮЧЕНИЯ	183
ГЛАВА 9. БИБЛИОТЕКА	201
ГЛАВА 10. ИМЕНОВАНИЕ И ВИДИМОСТЬ (НА ПРИМЕРЕ АДЫ)	205
ГЛАВА 11. ОБМЕН С ВНЕШНЕЙ СРЕДОЙ	219
ГЛАВА 12. ДВА АЛЬТЕРНАТИВНЫХ ПРИНЦИПА СОЗДАНИЯ ЯП	237
ЧАСТЬ II. ПЕРСПЕКТИВЫ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ	267

ГЛАВА 1. ПЕРСПЕКТИВНЫЕ МОДЕЛИ ЯЗЫКА	269
ГЛАВА 2. ФУНКЦИОНАЛЬНОЕ ПРОГРАММИРОВАНИЕ (МОДЕЛЬ Б)	289
ГЛАВА 3. ДОКАЗАТЕЛЬНОЕ ПРОГРАММИРОВАНИЕ (МОДЕЛЬ Д)	315
ГЛАВА 4. РЕЛЯЦИОННОЕ ПРОГРАММИРОВАНИЕ (МОДЕЛЬ Р)	339
ГЛАВА 5. ПАРАЛЛЕЛЬНОЕ ПРОГРАММИРОВАНИЕ В ОККАМЕ-2 (МОДЕЛЬ О)	353
ГЛАВА 6. НАСЛЕДУЕМОСТЬ (К ИДЕАЛУ РАЗВИТИЯ И ЗАЩИТЫ В ЯП)	391
ГЛАВА 7. ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ	415
ГЛАВА 8. ЗАКЛЮЧИТЕЛЬНЫЕ ЗАМЕЧАНИЯ	439
Заключение	459
Список литературы	460
Полезная литература, на которую прямых ссылок в тексте нет	463

Содержание

Предисловие ко второму изданию	14
Предисловие	15
Часть I. СОВРЕМЕННОЕ СОСТОЯНИЕ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ	19
Глава 1. Концептуальная схема языка программирования	21
1.1. Что такое язык программирования	22
1.2. Метауровень	22
1.3. Модель передачи сообщения	23
1.4. Классификация недоразумений	23
1.5. Отступление об абстракции-конкретизации. Понятие модели	25
1.6. Синтактика, семантика, прагматика	26
1.7. Зачем могут понадобиться знания о ЯП	27
1.8. Принцип моделирования ЯП	29
1.9. Пять основных позиций рассмотрения ЯП	29
1.10. Что такое производство программных услуг	30
1.11. Производство программных услуг – основная цель программирования	32
1.12. Сложность как основная проблема программирования	33
1.13. Источники сложности	34
1.14. Два основных средства борьбы со сложностью. Основной критерий качества ЯП	36
1.15. Язык программирования как знаковая система	37
1.16. Разновидности программирования	38
1.17. Понятие о базовом языке	39
1.18. Концептуальная схема рассмотрения ЯП	40
Глава 2. Пример современного базового ЯП (модель А)	43
2.1. Общее представление о ЯП Ада	44
2.2. Пример простой программы на Аде	46
2.3. Обзор языка Ада	47
2.3.1. Модули	48
2.3.2. Объявления и операторы	49
2.3.3. Типы данных	50
2.4. Пошаговая детализация средствами Ады	52
2.5. Замечания о конструктах	57
2.6. Как пользоваться пакетом управление_сетью	59

2.7. Принцип раздельного определения, реализации и использования услуг (принцип РОРИУС)	66
2.8. Принцип защиты абстракций	67
Глава 3. Важнейшие абстракции: данные, операции, связывание	69
3.1. Принцип единства и относительности трех абстракций	70
3.2. Связывание	71
3.3. От связывания к пакету	72
3.4. Связывание и специализация	74
3.4.1. Связывание и теория трансляции	75
3.5. Принцип цельности	79
3.5.1. Принцип цельности и нормальные алгоритмы	81
3.5.2. Принцип цельности и Ада. Критерий цельности	82
Глава 4. Данные и типы	85
4.1. Классификация данных	86
4.2. Типы данных	88
4.2.1. Динамические, статические и относительно статические ЯП	88
4.2.2. Система типов как знаковая система	90
4.2.3. Строгая типизация и уникальность типа	93
4.2.4. Критичные проблемы, связанные с типами	93
4.2.5. Критичные потребности и критичные языковые проблемы	94
4.2.6. Проблема полиморфизма	94
4.2.7. Янус-проблема	96
4.2.8. Критерий содержательной полноты ЯП. Неформальные теоремы	98
4.3. Регламентированный доступ и типы данных	98
4.3.1. Задача моделирования многих сетей	99
4.3.2. Приватные типы данных	101
4.3.3. Строго регламентированный доступ. Ограниченные приватные типы	103
4.3.4. Инкапсуляция	105
4.4. Характеристики, связанные с типом. Класс значений, базовый набор операций	106
4.5. Воплощение концепции уникальности типа. Определение и использование типа в Аде (начало)	107
4.5.1. Объявление типа. Конструктор типа. Определяющий пакет	107
4.6. Конкретные категории типов	108
4.6.1. Перечисляемые типы. «Морская задача»	108
4.6.2. Дискретные типы	116
4.6.3. Ограничения и подтипы	118
4.6.4. Квазистатический контроль	120
4.6.5. Подтипы	122

4.6.6. Принцип целостности объектов	123
4.6.7. Объявление подтипа	125
4.6.8. Подтипы и производные типы. Преобразования типа	125
4.6.9. Ссылочные типы (динамические объекты)	127
4.7. Типы как объекты высшего порядка. Атрибутные функции	129
4.7.1. Статическая определимость типа	129
4.7.2. Почему высшего порядка?	129
4.7.3. Действия с типами	129
4.8. Родовые (настраиваемые) сегменты	131
4.9. Числовые типы (модель числовых расчетов)	133
4.9.1. Суть проблемы	133
4.9.2. Назначение модели расчетов	134
4.9.3. Классификация числовых данных	134
4.9.4. Зачем объявлять диапазон и точность	135
4.9.5. Единая модель числовых расчетов	135
4.9.6. Допустимые числа	136
4.10. Управление операциями	137
4.11. Управление представлением	138
4.12. Классификация данных и система типов Ады	141
4.13. Предварительный итог по модели А	143
Глава 5. Раздельная компиляция	145
5.1. Понятие модуля	146
5.2. Виды трансляций	146
5.3. Раздельная трансляция	146
5.4. Связывание трансляционных модулей	147
5.4.1. Модули в Аде	147
5.5. Принцип защиты авторского права	148
Глава 6. Асинхронные процессы	151
6.1. Основные проблемы	152
6.2. Семафоры Дейкстры	155
6.3. Сигналы	157
6.4. Концепция внешней дисциплины	159
6.5. Концепция внутренней дисциплины: мониторы	160
6.6. Рандеву	164
6.7. Проблемы рандеву	165
6.8. Асимметричное рандеву	166
6.9. Управление асимметричным рандеву (семантика вспомогательных конструкторов)	167
6.10. Реализация семафоров, сигналов и мониторов посредством асимметричного рандеву	169
6.11. Управление асинхронными процессами в Аде	172

Глава 7. Нотация	175
7.1. Проблема знака в ЯП	176
7.2. Определяющая потребность	176
7.3. Основная абстракция	177
7.4. Проблема конкретизации эталонного текста	177
7.5. Стандартизация алфавита	178
7.6. Основное подмножество алфавита	179
7.7. Алфавит языка Ада	179
7.8. Лексемы	180
7.9. Лексемы в Аде	181
Глава 8. Исключения	183
8.1. Основная абстракция	184
8.2. Определяющие требования	185
8.3. Аппарат исключений в ЯП	187
8.3.1. Определение исключений	187
8.3.2. Распространение исключений. Принцип динамической ловушки	189
8.3.3. Реакция на исключение – принципы пластыря и катапульты	191
8.3.4. Ловушка исключений	193
8.4. Дополнительные особенности обработки исключений	194
Глава 9. Библиотека	201
9.1. Структура библиотеки	202
9.2. Компилируемый (трансляционный) модуль	202
9.3. Порядок компиляции и перекомпиляции (создания и модификации программной библиотеки)	203
9.4. Резюме: логическая и физическая структуры программы	204
Глава 10. Именованье и видимость (на примере Ады) ...	205
10.1. Имя как специфический знак	206
10.2. Имя и идентификатор	206
10.3. Проблема видимости	206
10.4. Аспекты именованья	207
10.5. Основная потребность и определяющие требования	207
10.6. Конструкты и требования, связанные с именованьем	208
10.7. Схема идентификации	210
10.7.1. Виды объявлений в Аде	210
10.7.2. Области локализации и «пространство имен» Ада-программы	213
10.7.3. Область непосредственной видимости	215
10.7.4. Идентификация простого имени	216
10.7.5. Идентификация составного имени	216
10.8. Недостатки именованья в Аде	216

Глава 11. Обмен с внешней средой	219
11.1. Специфика обмена	220
11.2. Назначение и структура аппарата обмена	223
11.2.1. Файловая модель	224
11.3. Файловая модель обмена в Аде	224
11.3.1. Последовательный обмен	225
11.3.2. Комментарий	226
11.3.3. Пример обмена. Программа диалога	229
11.3.4. Отступление о видимости и родовых пакетах	231
11.4. Программирование специальных устройств	233
Глава 12. Два альтернативных принципа создания ЯП ...	237
12.1. Принцип сундука	238
12.2. Закон распространения сложности ЯП	238
12.3. Принцип чемоданчика	239
12.4. Обзор языка Модуля-2	239
12.4.1. Характеристика Модуля-2 в координатах фон-неймановского языкового пространства (технологическая позиция)	240
12.5. Пример М-программы	241
12.5.1. Управление сетями на Модуле-2	242
12.5.2. Определяющий модуль	242
12.5.3. Используемый модуль	244
12.5.4. Реализующий модуль	245
12.6. Языковая ниша	248
12.7. Принцип чемоданчика в проектных решениях ЯП Модуля-2	249
12.7.1. Видимость	249
12.7.2. Инкапсуляция	251
12.7.3. Обмен	251
12.8. Принцип чайника	257
12.9. ЯП Оберон	258
12.9.1. От Модуля-2 к Оберону	259
12.9.2. Управление сетями на Обероне	261
Часть II. ПЕРСПЕКТИВЫ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ	267
Глава 1. Перспективные модели языка	269
1.1. Введение	270
1.2. Операционное программирование – модель фон Неймана (модель Н)	271
1.3. Ситуационное программирование – модель Маркова-Турчина (модель МТ)	273
1.3.1. Перевод в польскую инверсную запись (ПОЛИЗ)	274
1.3.2. Модификации модели Маркова (введение в Рефал)	275

1.3.3. Исполнитель (МТ-машина)	280
1.3.4. Программирование в модели МТ	280
1.3.5. Основное семантическое соотношение в модели МТ	281
1.3.6. Пример вычисления в модели МТ	283
1.3.7. Аппликативное программирование	285
1.3.8. Структуризация поля определений. МТ-функции	286
Глава 2. Функциональное программирование (модель Б)	289
2.1. Функциональное программирование в модели МТ	290
2.1.1. Модель МТ с точки зрения концептуальной схемы	290
2.1.2. Модель МТ и Лисп	291
2.1.3. Критерий концептуальной ясности и функции высших порядков ...	291
2.1.4. Зачем нужны функции высших порядков	292
2.1.5. Примеры структурирующих форм	294
2.1.6. Пример программы в стиле Бэкуса	297
2.2. Функциональное программирование в стиле Бэкуса (модель Б)	299
2.2.1. Модель Бэкуса с точки зрения концептуальной схемы	299
2.2.2. Объекты	300
2.2.3. Аппликация	300
2.2.4. Функции	301
2.2.5. Условные выражения Маккарти	301
2.2.6. Примеры примитивных функций	302
2.2.7. Примеры форм, частично известных по работе в модели МТ	304
2.2.8. Определения	306
2.2.9. Программа вычисления факториала	306
2.2.10. Программа перемножения матриц	308
Глава 3. Доказательное программирование (модель Д)	315
3.1. Зачем оно нужно	316
3.2. Доказательное программирование методом Бэкуса	316
3.2.1. Алгебра программ в модели Б	317
3.2.2. Эквивалентность двух программ перемножения матриц	318
3.3. Доказательное программирование методом Хоара	321
3.3.1. Модель Д	322
3.3.2. Дедуктивная семантика	324
3.3.3. Компоненты исчисления Хоара	326
3.3.4. Правила преодоления конструкторов языка Д	328
3.3.5. Применение дедуктивной семантики	334
Глава 4. Реляционное программирование (модель Р)	339
4.1. Предпосылки	340
4.2. Ключевая идея	341

4.3. Пример	341
4.3.1. База данных	342
4.3.2. База знаний	342
4.3.3. Пополнение базы данных (вывод фактов)	343
4.3.4. Решение задач	344
4.3.5. Управление посредством целей	344
4.4. О предопределенных отношениях	347
4.5. Связь с моделями МТ и Б	348
4.5.1. Путь от модели Б	348
4.5.2. Путь от модели МТ	350

Глава 5. Параллельное программирование в Оккаме-2 (модель О)

5.1. Принципы параллелизма в Оккаме	354
5.2. Первые примеры применения каналов	355
5.3. Сортировка конвейером фильтров	356
5.4. Параллельное преобразование координат (умножение вектора на матрицу)	357
5.4.1. Структура коллектива процессов	358
5.4.2. Коммутация каналов	361
5.5. Монитор Хансена-Хоара на Оккаме-2	362
5.6. Сортировка деревом исполнителей	363
5.7. Завершение работы коллектива процессов	366
5.8. Сопоставление концепций параллелизма в Оккаме и в Аде	368
5.8.1. Концепция параллелизма в Аде	369
5.8.2. Параллельное преобразование координат в Аде	371
5.9. Перечень неформальных теорем о параллелизме в Аде и Оккаме ...	377
5.10. Единая модель временных расчетов	378
5.11. Моделирование каналов средствами Ады	378
5.12. Отступление о задачных и подпрограммных (процедурных) типах	381
5.12.1. Входовые типы – фрагмент авторской позиции	381
5.12.2. Обоснование входовых типов	384
5.12.3. Родовые подпрограммные параметры	386
5.12.4. Почему же в Аде нет подпрограммных типов?	387
5.12.5. Заключительные замечания	387

Глава 6. Наследуемость (к идеалу развития и защиты в ЯП)

6.1. Определяющая потребность	392
6.2. Идеал развиваемости	392
6.3. Критичность развиваемости	393
6.4. Аспекты развиваемости	393
6.5. Идеал наследуемости (основные требования)	395
6.6. Проблема дополнительных атрибутов	395

6.7. Развитая наследуемость	398
6.8. Аспект данных	398
6.9. Аспект операций	401
6.10. Концепция наследования в ЯП (краткий обзор)	407
6.10.1. Основные понятия и неформальные аксиомы наследования	407
6.11. Преимущества развитой наследуемости	409
6.12. Наследуемость и гомоморфизм (фрагмент математической позиции)	410

Глава 7. Объектно-ориентированное программирование

7.1. Определяющая потребность	416
7.2. Ключевые идеи объектно-ориентированного программирования ...	417
7.3. Пример: обогащение сетей на Турбо Паскале 5.5	418
7.4. Виртуальные операции	423
7.5. Критерий Дейкстры	431
7.6. Объекты и классы в ЯП Симула-67	432
7.7. Перспективы, открываемые объектной ориентацией средств программирования	434
7.8. Свойства объектной ориентации	437
7.9. Критерий фундаментальности языковых концепций	438

Глава 8. Заключительные замечания

8.1. Реализаторская позиция	440
8.1.1. Компоненты реализации	440
8.1.2. Компиляторы	443
8.1.3. Основная функция компилятора	444
8.1.4. Три принципа создания компиляторов	445
8.2. Классификация языков программирования	448
8.2.1. Традиционная классификация	448
8.2.2. Недостатки традиционной классификации	450
8.2.3. Принцип инерции программной среды	450
8.2.4. Заповеди программиста	451
8.3. Тенденции развития ЯП	451
8.3.1. Перспективные абстракции	451
8.3.2. Абстракция от программы (в концептуальном и реляционном программировании)	455
8.3.3. Социальный аспект ЯП	457
8.3.4. Стандартизация ЯП	457

Заключение

Список литературы

Полезная литература, на которую прямых ссылок в тексте нет

*Моим дорогим родителям
Александре Фоминичне Каревой
Шахно Мордуховичу Кауфману
Эдит Яковлевне Кауфман*

Предисловие ко второму изданию

К немалому удивлению автора, эта книга оказалась востребованной и через 17 лет после своего официального выхода в свет (насколько известно автору, ее используют в МГУ, МАИ и других российских университетах). Это тем более удивительно, что основной ее материал подготовлен значительно раньше, примерно в 1985 году.

В Сети также циркулируют (и даже продаются, якобы с разрешения автора ☺) от-носительно ранние варианты лекций по курсу «Языки программирования», читанных автором в те же годы на факультете ВМиК МГУ.

Многие уважаемые члены программистского сообщества посчитали нужным под-держать уверенность автора в ценности изложенного в книге материала.

Владимир Ильич Головач в своей рецензии в «Мир ПК» одним из первых предска-зал ей долгую жизнь. Андрей Андреевич Терехов, известный знаток компьютерной литературы, также высоко оценил качество книги. Очень хорошо отзывались о ней Владимир Арнольдович Биллиг, Руслан Петрович Богатырев, Лев Николаевич Чер-нышов, а также многие другие. Всем этим людям огромная благодарность за поддерж-ку и стимулирование настоящего издания.

Книга оказалась также в перечне основной литературы, рекомендованной Минобра-зованием России для специальности 010200 «Прикладная математика и информатика».

Немало замечательных членов программистского сообщества, упоминаемых в кни-ге, многих из которых автор имел удовольствие знать лично и даже обсуждать с ними фрагменты книги или читавшегося в МГУ курса, за прошедшие годы покинули этот мир. Среди них Евгений Андреевич Жоголев, Александр Владимирович Замулин, Михаил Романович Шура-Бура, Владимир Федорович Турчин. Бесконечная им при-знательность за бесценный вклад в общее дело и светлая память.

Переиздание книги, несмотря на имеющиеся запросы, совсем недавно представля-лось совершенно нереальным с учетом стопроцентной загрузки автора основной рабо-той. Ведь электронная верстка книги оказалась утраченной в результате непростых пертурбаций бурных 90-х прошлого века.

К счастью, мир полон чудес. Одно из них – воскрешение полной электронной вер-сии книги с помощью современных средств сканирования с печатного оригинала и последующего ручного редактирования. Эта огромная работа была выполнена цели-ком по собственной инициативе Ольгой Львовной Бондаренко. Как по волшебству, подоспело и предложение Дмитрия Алексеевича Мовчана переиздать книгу в «ДМК Пресс».

Автору оставалось только вычитать полученный от Ольги Львовны Word-доку-мент и передать результат в «ДМК Пресс» в надежде помочь людям, желающим глу-боко вникнуть в суть проблем, принципов и концепций современного программиро-вания.

Удачи, глубокоуважаемый читатель!

В. Ш. Кауфман
12.04.10 Москва–Хельсинки

Предисловие

Эта книга возникла из курса лекций «Языки программирования», читаемого автором в МГУ. Стимулом для написания книги послужило отсутствие доступной литературы, в которой были бы систематически изложены, во-первых, ключевые принципы, концепции и понятия, составляющие основу предмета и поэтому претендующие на относительную стабильность, и, во-вторых, перспективные идеи и тенденции, помогающие ориентироваться в огромном и быстро меняющемся мире современных языков программирования (ЯП).

Автор столкнулся с немалыми проблемами, несмотря на то что становление современных ЯП происходило, можно сказать, на его глазах. Пришлось пережить и восторги от изобретения первых «языков высокого уровня», быть среди тех, кто увлекался их «усовершенствованием» и созданием первых трансляторов, опираясь только на здравый смысл и собственную смекалку, пришлось пережить и надежды на создание «универсального ЯП» объединенными усилиями международного программистского братства, и разочарования от бездарной траты сил и средств на бесперспективные начинания.

Когда в результате преодоления части этих проблем выяснилось, что удастся существенно прояснить суть дела (частично отбирая, частично изобретая принципы, концепции и понятия), фрагменты книги, к удивлению автора, оказались интересны не только студентам и коллегам-преподавателям, но и программистам – профессионалам и специалистам по ЯП. По-видимому, проблемы, с которыми столкнулся автор, осмысливая один из важнейших аспектов информатики, оказались жизненно важными для существенно более широкого круга потенциальных читателей, а отражение опыта их преодоления в тексте книги – достаточно интересным и поучительным.

Заметив это обстоятельство, автор уже сознательно стал иногда рассчитывать не только на студенческую аудиторию, но и на более искушенного читателя, позволяя себе намеки и аналогии, подразумевающие личный опыт программирования и даже экспертной деятельности в области ЯП. Более того, стало очень трудно отделить то, что известно, признано, устоялось, от того, что удалось только что понять, систематизировать, придумать. В результате жанр книги стал менее определенным, «поплыл» от первоначально задуманного учебного пособия в сторону монографии.

С точки зрения ключевых концепций и принципов, определяющих современное состояние и перспективы в области ЯП, конкретные ЯП интересны не сами по себе, а прежде всего как источники примеров при обсуждении излагаемых положений. Поэтому систематически применяется метод моделирования ЯП – изучаются не ЯП в целом, а только их модели. Конечно, автор старался дать необходимый минимум сведений о ЯП, позволяющий понимать написанные на нем примеры без привлечения дополнительной литературы. В качестве основного метода знакомства с ЯП в книге принят метод «погружения», столь популярный при ускоренном обучении иностранным языкам, – сведения о ЯП читателю предлагается извлекать непосредственно из примеров, написанных на этом ЯП программ (естественно, с подробными комментариями). Опыт показывает, что такой путь обеспечивает достижение основной цели с приемлемыми затратами времени и сил. Поэтому в книге нет подробных описаний конкретных ЯП – желающие могут воспользоваться официальными сообщениями, фирменной документацией или учебниками по ЯП.

Немыслимо в одной книге содержательно обсудить все (даже только важнейшие) концепции и принципы, определяющие современные ЯП. Пришлось выработать критерий отбора. Он касается и проблем программирования в целом, и назначения ЯП, и их выбираемых для обсуждения свойств. Из всех проблем программирования в качестве ключевой выбрана проблема сложности (самих программ, их создания, средств их создания и т. п.). Основным источником сложности считается семантический разрыв – рассогласование моделей мира у потенциального пользователя и потенциального исполнителя программ (компьютера). В качестве основных средств преодоления этого разрыва выделены, с одной стороны, аппарат абстракции-конкретизации (аппарат развития), а с другой – аппарат прогнозирования-контроля (аппарат защиты). Основной объект изучения – это концепции, принципы и понятия, позволяющие строить концептуально-целостные ЯП с мощным аппаратом развития и надежной защитой.

Книга состоит из двух частей. Первая посвящена основным абстрациям, используемым в современных ЯП. В качестве основного языка примеров здесь фигурирует ЯП Ада. Он удобен в этой роли потому, что в той или иной форме содержит ответы практически на все технологические проблемы. Другими словами, Ада служит примером «максимального» современного ЯП. «Минимальные» ЯП представлены языками Никлауса Вирта – это Модула-2 и Оберон (образца 1988 г.).

Вторая часть рассказывает о перспективных тенденциях в ЯП. В ней рассмотрены ситуационное, функциональное, доказательное, реляционное, параллельное и объектно-ориентированное программирование. Среди языков-примеров – Рефал, функциональный язык Бэкуса, Оккам-2 для программирования транспьютеров, объектно-ориентированный Турбо Паскаль и др.

В книге немало вопросов и упражнений (снабженных обычно подсказками), призванных помочь читателю управлять своим вниманием и контролировать уровень усвоения материала. Результаты упражнений при дальнейшем изложении не используются.

Так как замысел книги возник восемь лет назад и почти половина материала написана еще в 1983–1985 гг., закономерно опасение, не устарела ли книга еще до своего выхода в свет. Конечно, судить об этом читателю, однако автор старался отбирать фундаментальные и, по его мнению, перспективные концепции и принципы, которые по самой своей природе должны быть стабильнее быстро меняющейся конъюнктуры.

Косвенным подтверждением такой стабильности послужил весьма обрадовавший автора факт, что современный всплеск (своеобразный «бум») интереса к объектно-ориентированному программированию – это в сущности всплеск интереса к средствам программирования, обеспечивающим рациональное развитие программных услуг при надежной защите авторского права. Но именно средства развития и защиты в ЯП и были выбраны в качестве самого интересного аспекта ЯП еще в начале работы над книгой. Такое знаменательное совпадение придает уверенности в правильности выбора и позволяет считать объектно-ориентированное программирование не просто очередной модой, а естественной «закрывающей скобкой» как очередного этапа в осознании системы ценностей в программировании, так и нашего рассмотрения концепций и принципов ЯП.

Еще одним подтверждением тезиса о фундаментальности рассматриваемых в книге концепций и принципов может служить тот факт, что как в разработанных в конце

1990 г. требованиях на создание обновленного международного стандарта языка программирования Ада (учитывающих самые современные пожелания интернационального сообщества пользователей языка Ада и самые современные методы их удовлетворения), так и в аванпроекте обновленного языка, датированном февралем 1991 г., нашли отражение, кроме объектно-ориентированного программирования, и такие рассмотренные в книге проблемы, как развитие концепции управления асинхронными процессами, развитие концепции типа, развитие концепции исключений и др.

Создавать эту книгу помогали многие люди, которые, конечно, не несут какой-либо ответственности за ее недостатки. Автору приятно выразить признательность В. К. Мережкову за инициативу и огромную помощь при издании первой части рукописи в НПО «Центрпрограммсистем», профессорам Е. Л. Ющенко, М. Р. Шура-Буре, В. Н. Редько, И. М. Витенбергу, А. А. Красиллову, С. С. Лаврову, Я. Я. Осису, Е. А. Жоголеву, Н. П. Трифонову, Г. С. Цейтину за поддержку и ценные советы, своим коллегам и первым читателям В. Л. Темову, В. Н. Агафонову, В. И. Головачу, А. С. Маркову, Б. Г. Чеблакову, Анд. В. Климову, В. Н. Лукину, И. В. Раковскому за содержательную критику, А. Л. Александрову, И. Н. Зейтленок, И. З. Луговой и особенно С. И. Рыбину за помощь в подготовке рукописи, своим слушателям и студентам за внимание, терпение и любознательность, своим родным – за понимание и заботу.

Автор старался не изменить духу преданности сути дела и творческой раскованности, воплощением которых для него остаются рано ушедшие из жизни Андрей Петрович Ершов, успевший прочитать первый вариант рукописи и поддержать настроение автора писать «как пишется», и Адольф Львович Фуксман, который в свое время горячо обсуждал с В. Л. Темовым и автором совместный проект университетского учебника по программированию.

В. Ш. Кауфман

Современное состояние языков программирования

Глава 1. Концептуальная схема языка программирования	21
Глава 2. Пример современного базового ЯП (модель А)	43
Глава 3. Важнейшие абстракции: данные, операции, связывание ...	69
Глава 4. Данные и типы	85
Глава 5. Раздельная компиляция	145
Глава 6. Асинхронные процессы	151
Глава 7. Нотация	175
Глава 8. Исключения	183
Глава 9. Библиотека	201
Глава 10. Именованное и видимость (на примере Ады) ..	205
Глава 11. Обмен с внешней средой	219
Глава 12. Два альтернативных принципа создания ЯП	237

Концептуальная схема языка программирования

1.1. Что такое язык программирования	22
1.2. Метауровень	22
1.3. Модель передачи сообщения	23
1.4. Классификация недоразумений	23
1.5. Отступление об абстракции-конкретизации. Понятие модели	25
1.6. Синтактика, семантика, прагматика	26
1.7. Зачем могут понадобиться знания о ЯП	27
1.8. Принцип моделирования ЯП	29
1.9. Пять основных позиций рассмотрения ЯП	29
1.10. Что такое производство программных услуг	30
1.11. Производство программных услуг – основная цель программирования	32
1.12. Сложность как основная проблема программирования	33
1.13. Источники сложности	34
1.14. Два основных средства борьбы со сложностью. Основной критерий качества ЯП	36
1.15. Язык программирования как знаковая система	37
1.16. Разновидности программирования	38
1.17. Понятие о базовом языке	39
1.18. Концептуальная схема рассмотрения ЯП	40

Конец ознакомительного фрагмента.

Приобрести книгу можно

в интернет-магазине

«Электронный универс»

e-Univers.ru