

УДК 004.43

ЯЗЫКИ ПРОГРАММИРОВАНИЯ: КЛАССИФИКАЦИЯ, ОСОБЕННОСТИ, КРИТЕРИИ ВЫБОРА

А.В. Пахунов*

Аннотация. Рассматриваются вопросы появления различных языков программирования, определения особенностей этих языков, а также составления основных видов и классификаций языков программирования. Предлагаются основные факторы и критерии выбора языка программирования для разработки программного обеспечения.

Ключевые слова: языки программирования, информационные технологии, классификации языков программирования, критерии выбора языка программирования.

PROGRAMMING LANGUAGES: CLASSIFICATION, FEATURES, CRITERIA OF CHOICE

A.V. Pakhunov

Abstract. This article is devoted to the questions of emerging different programming languages, identifying their features, as well as drawing up the main types of classifications for these programming languages. The article suggests the main factors and criteria for choosing programming language for developing any software.

Keywords: programming languages, information technology, classification of the programming languages, criteria for choosing any programming language.

Научная специальность 05.13.17 — Теоретические основы информатики

*«Язык формирует наш способ мышления
и определяет то, о чем мы можем мыслить».*

Б.Л. Уорф

Прогресс компьютерных технологий определил процесс появления новых разнообразных знаковых систем для записи алгоритмов — языков программирования. Язык программирования — язык (знаковая система), предназначенный для записи компьютерных программ; определяет набор лексических, синтаксических и семантических правил, определяющих внешний вид программы и действия, которые выполнит исполнитель (обычно — ЭВМ) под ее управлением.

Со времени создания первых программируемых машин человечество придумало более 8 тыс. языков программирования (включая нестандартные, визуальные и эзотерические языки) [1]. Каждый год их число увеличивается. Некоторыми языками умеет пользоваться только небольшое число их собственных разработчиков, другие становятся известны миллионам людей. Профессиональные программисты могут владеть десятком и более разных языков программирования.

Язык программирования предназначен для написания компьютерных программ, которые представляют собой набор правил, позволяющих компьютеру выполнить тот или иной вычислительный процесс, организовать управление различными объектами, и т.п. [3]. Язык программирования отличается от естественных языков тем, что предназначен для взаимодействия человека с ЭВМ, в то время как естественные языки используются для общения людей между собой.

Большинство языков программирования использует специальные конструкции для определения и манипулирования структурами данных и управления процессом вычислений.

В период 1960-х—1970-х гг. были разработаны основные парадигмы языков программирования, используемые в настоящее время, хотя во многих аспектах этот процесс представлял собой улучшение идей и концепций, заложенных еще в первых языках третьего поколения [1]:

- язык APL оказал влияние на функциональное программирование и стал первым языком, поддерживавшим обработку массивов;



- язык ПЛ/1 (NPL) был разработан в 1960-х гг. как объединение лучших черт Фортрана и Кобола;

- язык Симула, появившийся примерно в это же время, впервые включал поддержку объектно-ориентированного программирования;

- в середине 1970-х гг. группа специалистов представила язык Smalltalk, который был уже всецело объектно-ориентированным;

- в период с 1969 по 1973 гг. велась разработка языка Си, популярного и по сей день, и ставшего основой для множества последующих языков, например, столь популярных, как C++ и Java;

- в 1972 г. был создан Пролог — наиболее известный (хотя и не первый, далеко не единственный) язык логического программирования;

- в 1973 г. в языке ML была реализована расширенная система полиморфной типизации, положившая начало типизированным языкам функционального программирования.

Каждый из этих языков породил по семейству потомков, и большинство современных языков программирования, в конечном счете, основано на одном из них.

Кроме того, в 1960—1970-х гг. активно велись споры о необходимости поддержки структурного программирования в тех или иных языках [2]. Голландский специалист Э. Дейкстра выступал в печати с предложениями о полном отказе от использования инструкций GOTO во всех высокоуровневых языках. Развивались также приемы, направленные на сокращение объема программ и повышение продуктивности работы программиста и пользователя.

Существуют различные классификации языков программирования. По наиболее распространенной классификации все языки программирования, в соответствии с тем, в каких терминах необходимо описать задачу, делят на языки низкого и высокого уровня [4]. Если язык близок к естественному языку программирования, то он называется языком высокого уровня, если ближе к машинным командам — языком низкого уровня.

В группу языков низкого уровня входят машинные языки и языки символического кодирования: Автокод, Ассемблер. Операторы этого языка — это те же машинные команды, но записанные мнемоническими кодами, а в качестве операндов используются не конкретные адреса, а символические имена. Все языки низкого уровня ориентированы на определенный тип компьютера, т.е. являются машинно-зависимыми.

Машинно-ориентированные языки — это языки, наборы операторов и изобразительные средства которых существенно зависят от особенностей ЭВМ.

К языкам программирования высокого уровня относят Фортран (переводчик формул — был разработан в се-

редине 50-х гг. программистами фирмы IBM и в основном используется для программ, выполняющих естественно-научные и математические расчеты), Алгол, Кобол (коммерческий язык — используется, в первую очередь, для программирования экономических задач), Паскаль, Бейсик (был разработан профессорами Дармутского колледжа — Джоном Кемени и Томасом Курцом), Си (Деннис Ритч — 1972 г.), Пролог (в основе языка лежит аппарат математической логики) и т.д.

Эти языки машинно-независимы, так как они ориентированы не на систему команд той или иной ЭВМ, а на систему операндов, характерных для записи определенного класса алгоритмов. Однако программы, написанные на языках высокого уровня, занимают больше памяти и медленнее выполняются, чем программы на машинных языках. Программу, написанную на языке программирования высокого уровня, ЭВМ не понимает, поскольку ей доступен только машинный язык. Поэтому для перевода программы с языка программирования на язык машинных кодов используют специальные программы-трансляторы.

Существует три вида транслятора: интерпретаторы (это транслятор, который производит пооператорную обработку и выполнение исходного кода программы), компиляторы (преобразует всю программу в модуль на машинном языке, после чего программа записывается в память компьютера и лишь потом исполняется) и ассемблеры (переводят программу, записанную на языке ассемблера, в программу на машинном языке) [2].

Языки программирования также можно разделять на поколения [4]:

- языки первого поколения: машинно-ориентированные с ручным управлением памяти на компьютерах первого поколения;

- языки второго поколения: с мнемоническим представлением команд, так называемые автокоды;

- языки третьего поколения: общего назначения, используемые для создания прикладных программ любого типа. Например, Бейсик, Кобол, Си и Паскаль;

- языки четвертого поколения: усовершенствованные, разработанные для создания специальных прикладных программ, для управления базами данных;

- языки программирования пятого поколения: языки декларативные, объектно-ориентированные и визуальные. Например, Пролог, ЛИСП (используется для построения программ с использованием методов искусственного интеллекта), Си++, Visual Basic, Delphi.

Языки программирования также можно классифицировать на процедурные и неперечисленные [2]. В процедурных языках программа явно описывает действия, которые необходимо выполнить, а результат задается только способом получения его при помощи некоторой процедуры,

которая представляет собой определенную последовательность действий. Среди процедурных языков выделяют структурные и операционные языки. В структурных языках одним оператором записываются целые алгоритмические структуры: ветвления, циклы и т.д.; в операционных языках для этого используются несколько операций. Широко распространены следующие структурные языки: Паскаль, Си, Ада, ПЛ/1. Среди операционных — Фортран, Бейсик, Фокал.

Непроцедурное (декларативное) программирование появилось в начале 70-х гг. XX в. К непроцедурному программированию относятся функциональные и логические языки. В функциональных языках программа описывает вычисление некоторой функции. Обычно эта функция задается как композиция других, более простых, т.е. в свою очередь, делятся на еще более простые задачи и т.д. Один из основных элементов функциональных языков — рекурсия. Оператора присваивания и циклов в классических функциональных языках нет. В логических языках программа вообще не описывает действий. Она задает данные и соотношения между ними. После этого системе можно задавать вопросы. Машина перебирает известные и заданные в программе данные и находит ответ на вопрос. Порядок перебора не описывается в программе, а неявно задается самим языком. Классическим языком логического программирования считается Пролог. Программа на Прологе содержит, набор предикатов-утверждений, которые образуют проблемно-ориентированную базу данных и правила, имеющие вид условий.

Еще один класс языков программирования — это объектно-ориентированные языки высокого уровня. На таких языках не описывают подробной последовательности действий для решения задачи, хотя они содержат элементы процедурного программирования. Благодаря богатому пользовательскому интерфейсу, они предлагают человеку решить задачу в удобной для него форме [3]. Первый объектно-ориентированный язык программирования Simula был создан в 1960-х гг. Нигаардом и Далом.

Ява — язык для программирования Интернет, позволяющий создавать безопасные, переносимые, надежные, объектно-ориентированные интерактивные программы. Язык Ява жестко связан с Интернетом, потому что первой серьезной программой, написанной на этом языке, был браузер Всемирной паутины.

Говоря о программировании в Интернете, часто имеют в виду создание публикаций с использованием языка разметки гипертекстовых документов — HTML. Применение специальных средств (HTML-редакторов) позволяет не только создавать отдельные динамически изменяющиеся интерактивные HTML-документы, используя при этом данные мультимедиа, но и редактировать целые сайты.

Когда необходимо создать большую программную систему или составить программы для решения какой-либо частной задачи, в том числе при создании компьютерных игр, встает вопрос, какой выбрать для этой цели подходящий язык программирования. Такой выбор делается на основании очень простых «земных» факторов: наличия того или иного транслятора и умения писать программы на данном языке. Но если в распоряжении пользователя несколько языков программирования и нужно создать, например, игру для мобильного, то необходимо учитывать следующие обстоятельства [3]:

- назначение разрабатываемого программного обеспечения, т.е. необходимость в нем будет временная или оно будет использоваться постоянно, будет ли оно в дальнейшем передаваться другим организациям, будут ли создаваться его новые версии;
- необходимая скорость работы программного обеспечения, соотношение и работа его вычислительных и диалоговых компонентов;
- предполагаемый размер программы, т.е. создавать ее как единое целое, или она будет в виде отдельных взаимодействующих модулей, нужно ли минимизировать размер памяти, которую занимает программа во время работы;
- возможность сопряжения разрабатываемого программного обеспечения с другими приложениями (пакетами или программами), включая приложения, составленные на иных языках программирования;
- основные типы данных, которыми придется оперировать, возможность поддержки работы с различными типами структур;
- характер и уровень использования периферийных средств (монитора, клавиатуры и др.), необходимость в специальном программировании некоторых функций, чтобы работать с периферийными устройствами;
- целесообразность и возможность применения имеющихся стандартных библиотек подпрограмм, процедур, функций.

Литература

1. Бьянкуцци Ф., Уорден Ш. Пионеры программирования. Диалоги с создателями наиболее популярных языков программирования. СПб., 2010.
2. Вольфенгаген В.Э. Конструкции языков программирования. Приемы описания. М., 2001.
3. Паронджанов В.Д. Как улучшить работу ума. Алгоритмы без программистов — это очень просто! М., 2001.
4. Себеста Р.У. Основные концепции языков программирования = Concepts of Programming Languages / Пер. с англ.; 5-е изд. М., 2001.