

Виды языков программирования

Лекция 4

Парадигмы программирования

Парадигма — это главная идея какого-либо сложного понятия (в данном случае языка программирования).

Применительно к языкам программирования различают следующие парадигмы:

- **императивное программирование;**
- **функциональное программирование;**
- **логическое программирование;**
- **объектно-ориентированное программирование.**

Императивные языки программирования

Императивные (процедурные) языки программирования (ЯП) задают вычисления как последовательность команд (операторов).

Они ориентированы на компьютеры с архитектурой фон Неймана.

Основные понятия императивных ЯП тесно связаны с компонентами компьютера:

- переменные различных типов (моделируют ячейки памяти);
- операторы присваивания (моделируют пересылки данных);
- повторения действий в форме итерации (моделируют хранение информации в смежных ячейках памяти).

ИЯ – состояние компьютера

Состояние компьютера — это множество всех значений всех ячеек его памяти. Программа состоит из последовательности операторов, выполнение каждого из которых влечет за собой изменение значения в одной или нескольких ячейках памяти, то есть переход компьютера в новое состояние. В общем случае синтаксис императивной программы имеет вид:

- оператор1;
- оператор2;
- ...

Императивные языки

- Fortran (1954, Д. Бэкус),
- Algol 60 (1960, П. Наур),
- Cobol (1960, Пентагон),
- PL/1 (1965, IBM),
- Pascal (1971, Н. Вирт),
- C (1972, Д. Ритчи).

Язык Фортран

- Первый язык программирования высокого уровня, который получил широкое признание
- Создан фирмой IBM в 1957 году
- В языке Fortran задано сравнительно мало типов данных: четыре типа числовых данных (целые, вещественные и комплексные числа, а также вещественные с двойной точностью), *булевы* данные, массивы, строки символов и файлы. Поскольку язык предназначен для научных вычислений, в нем имеется большое количество встроенных математических функций и арифметических операций.

Язык C

- Язык C был разработан в 1972 году Дэнисом Ритчи и Кеном Томпсоном из лаборатории AT&T Bell Telephone Laboratories. По стилю он похож на Algol и Pascal. Язык C — небольшой, но универсальный язык с ограниченным набором операторов и возможностей. Вместе с тем, органичный сплав простого синтаксиса и высокой эффективности вычислений принесли ему славу самого популярного языка системного программирования.
- Использован при создании операционной системы UNIX

Понятия введенные в языке C

- *Препроцессор C.* Каждый компилятор C включает в себя директивы препроцессора, начинающиеся с символа #, которые не являются составной частью самого языка C.
- *Соглашения интерфейса C.* В результате применения языка C возник ряд соглашений. Например, принято, что определения интерфейсов между модулями должны храниться в соответствующем заголовочном файле с расширением .h.
- *Библиотека C.* Многие функции (printf, getchar, malloc, fork и exec) были написаны с помощью интерфейсов C, хотя они и не входят в официальное описание языка C. Однако в стандарте языка C эти функции уже включены в язык (как обязательные библиотечные функции для стандартных компиляторов). Подключение большой библиотеки позволяет существенно расширять функциональные возможности ядра языка.

Функциональные языки программирования

- Вместо отслеживания последовательности состояний, через которые должен пройти компьютер для получения ответа, здесь ищется функция программы, обеспечивающая ожидаемый результат.
- **Функциональные (аппликативные) языки задают вычисления как вызовы функций**
- Синтаксис функциональной программы выглядит следующим образом:

$function_n (...function_2(function_1(данные))...)$

Особенности функционального языка (ФЯ)

- В ФЯ отсутствуют переменные , нет операторов присваивания, нет повторений в форме итераций
- Чтобы определить функцию нужно придумать ее имя и задать вычисляемое ей выражение.
- Описание единичного действия – это вызов функции с конкретными аргументами
- Описание программы – это описание последовательности вызовов отдельных функций

Преимущества функционального программирования

- Прежде всего, функциональные программы предлагают предельно естественную, математическую форму записи вычислений.
- Программы получаются короткими, выразительными и понятными. Здесь доминирует более высокий уровень программирования, требующий меньших усилий, чем императивное программирование.
- Программист свободен от необходимости учитывать те детали, которыми наполнена императивная программа.

Проблемы реализации ФЯ

- В функциональном языке программисту не нужно связываться с переменными, поскольку здесь не требуются такие абстракции, как ячейки памяти.
- Для организации повторения действий используется мощный механизм рекурсии, тоже оставляющий в стороне детали реализации.
- Все это выливается в большие накладные расходы времени и памяти при выполнении программы.
- Компьютер с последовательным порядком работы и линейной организацией памяти просто не способен обеспечить высокую эффективность функциональных вычислений.

Функциональные языки

- Первый язык функционального программирования LISP появился в рамках проекта по искусственному интеллекту (1960). Разработан под руководством Джона Маккарти из Массачузетского технологического института
- Язык Miranda (1986) предложил Дэвид Тернер из университета Кента.
- Язык ML (MetaLanguage) создан Робинот Милнером в Эдинбургском университете (1990). Поддерживает не только функциональную, но и императивную парадигму.
- Язык Haskell (1992) чисто функциональный язык - современный преемник языка Miranda

Особенности языка LISP

- Эквивалентность форм представления в языке программ и данных, что позволяет выполнять структуры данных как программы и изменять программы как данные.
- В качестве структуры управления применяется рекурсия, а не итерация (цикл), как в императивных языках. В большинстве программ на LISP широко используются рекурсивные вызовы функций
- Проще всего LISP реализуется с помощью программного интерпретатора и программного моделирования всех элементарных операций.

Логические языки программирования

- **Логические (декларативные) языки описывают не способ решения задачи, а саму задачу.** Фактически они задают формализованное представление предметной области.
- Иногда логические языки называют *языками, основанными на системе правил*. Здесь осуществляется проверка некоторого разрешающего высказывания (утверждения), и в случае его истинного значения выполняется соответствующее действие.
- В языке, основанном на системе правил, не существует определенного порядка применения правил, и система реализации языка должна сама выбрать нужный порядок выполнения действий, который приведет к желаемому результату.

Логическая машина имеет

набор фактов

любит (том, футбол)

любит (боб, баскетбол)

любит (ник, теннис)

аксиомы (правила вывода)

любит (том, X) : – любит (билл, X)

Имена постоянных пишутся с маленькой буквы (том), а имена переменных – с большой (X).

Программа – это запрос к логической машине

? любит (билл, Y)

Работа логической машины

- Цель выполнения программы: замена переменной на постоянную. Для этого просматриваются факты и аксиомы, ищутся возможные подстановки. Они могут выполняться неоднократно.
 - Подстановки для нашей программы:
 1. любит (том, Y)
 2. любит (том, футбол)
- В результате вместо Y подставляется футбол.

Языки логического программирования

- Язык Prolog. Создан авторским коллективом из преподавателей Марсельского и Эдинбургского университетов (Франция, 1972).
- В 1981 году Япония инициировала государственный проект по созданию компьютеров нового поколения, рабочим языком которых должен был стать Prolog. Проект провалился (из-за ограниченных возможностей существующей элементной базы).

Объектно-ориентированные языки программирования

Парадигма объектно-ориентированного программирования является развитием императивного программирования. При ее создании преследовались две цели:

- сократить размеры программ за счет повышения размера строительных элементов («маленькие» переменные заменяются «большими» объектами) и тем самым обеспечить возможность создания (за то же время) более крупных программных приложений;
- упростить процесс создания новых программ на базе старых (за счет применения механизма наследования).

Основные понятия ООП

Объектно-ориентированные языки задают вычисления как взаимодействия программных объектов.

- *Объект* — это именуемый модуль, заключающий в себе данные и операции для их обработки.
- Программный объект во многом похож на физический объект реального мира. В частности, программный объект имеет свое состояние и демонстрирует окружающей среде свое поведение.
- Состояние объекта характеризуется перечнем данных и их значений. Поведение задается последовательностью выполняемых операций.

Взаимодействие объектов

- Объекты взаимодействуют друг с другом с помощью сообщений. Посылается сообщение объектом-источником в адрес объекта-приемника. Каждое сообщение — это запрос на выполнение операции объектом-приемником.
- Описание объектов с общей структурой и поведением называется *классом*. Как и переменные, единичные объекты создаются по их описаниям. Только в роли описаний для переменных выступают типы данных, а в роли описаний для объектов — классы. Объект считается экземпляром класса

Принципы ООП

- инкапсуляция (сокрытие своего содержимого от внешнего мира);
- наследование (возможность получения потомками структуры и поведения предков);
- полиморфизм (использование одного и того же имени для выражения различных действий и объектов).

Языки ООП

- Первый объектно-ориентированный ЯП, Simula 67, был придуман норвежцами К. Нигаардом и У. Далом как расширение императивного языка Algol 60. Этот язык опередил свое время и был забыт.
- Второй объектно-ориентированный ЯП, Smalltalk, появился в нужное время (1972–980) и в нужном месте. Его автором стал Алан Кей из фирмы Xerox.
- Широкое распространение получил язык C++, созданный Б. Страуструпом (1983).

Язык HTML

- Язык HTML (HyperText Markup Language — язык разметки гипертекста) предназначен для создания гипертекстовых документов в среде WWW (World Wide Web — Всемирная паутина). HTML-документы (иначе веб-документы, или веб-страницы) могут просматриваться различными типами веб-браузеров. HTML — один из наиболее простых языков создания веб-страниц.
- У каждого компьютера в Интернете имеется свой адрес. Когда на веб-сервер приходит запрос, адрес отправителя всегда известен. Ответ посылается именно по этому адресу. Обеспечивают взаимодействие сервера и клиента несколько протоколов передачи данных

Протокол TCP/IP (Transmission Control Protocol/Internet Protocol).

После установления соединения по TCP/IP запускается веб-браузер, использующий протокол передачи гипертекста (http, hyper text transfer protocol) для организации обмена данными. Веб-сервер посылает HTML-документы на браузер клиента, который выводит их на экран. У каждой веб-страницы есть свой адрес. Эти адреса называются унифицированными указателями информационных ресурсов URL (Uniform Resource Locators). Например, URL для некоторого веб-документа на сайте корпорации Microsoft имеет вид:
<http://www.microsoft.com/windows/index.html>

Структура документов в языке HTML

- Описания документов в языке HTML создаются с помощью элементов.
- Элемент — это команда, записываемая тегами. Говорят, что элементы задают разметку текста документа.
- В общем случае элемент имеет следующую структуру:
Открывающий_тег Содержание Закрывающий_тег

Содержание документа

- Содержание — это некоторая последовательность символов.
- Все теги начинаются с левой угловой скобки (<) и заканчиваются правой угловой скобкой (>). Внутри угловых скобок указывается имя тега и, в случае открывающего тега, атрибуты.
- Имя открывающего тега формирует имя всего элемента. Имя открывающего тега повторяется в закрывающем теге, но предваряется символом косой черты (/)

Синтаксис документа HTML

Минимальный документ HTML имеет следующий синтаксис:

```
<html>
```

```
  <head>
```

```
    <title> Заглавие документа </title>
```

```
  </head>
```

```
  <body> Текст документа
```

```
</body>
```

```
</html>
```

Скриптовые языки

- Традиционные языки программирования предназначены прежде всего для создания автономных приложений: программ, которые принимают входные данные, обрабатывают их определенным образом и генерируют соответствующие результаты. Однако в большинстве компьютерных приложений требуется координация множества программ
- Сценарии подобных действий принято называть **скриптами**. Все эти сценарии обосновывают потребность в программах, которые координируют другие программы.

Применение скриптовых языков

- Универсальные скриптовые языки, такие как Perl и Python, иногда называют *склеивающими языками*, поскольку они создавались для «склеивания» существующих программ в интересах создания большой системы. С ростом Всемирной паутины скриптовые языки получили новый толчок в направлении генерации динамического информационного наполнения сети. Они также широко используются как *языки расширения*, которые позволяют пользователю настраивать или расширять функциональные возможности «скриптовых» утилит.

Общая характеристика скриптовых языков

- Пакетное и интерактивное использование
- Экономичность выражений
- Отсутствие объявлений, простые правила видимости данных
- Гибкая динамическая типизация
- Легкий доступ к другим программам
- Сложное сопоставление с образцами и обработка строк
- Высокоуровневые типы данных

Язык PHP

- Создатель языка Расмус Лерддорф (1994)
- PHP «PHP Hypertext Preprocessor». Язык PHP сейчас разрабатывается, распространяется и поддерживается как продукт с открытым исходным кодом. Процессоры PHP располагаются на большинстве веб-серверов. PHP является скриптовым языком серверной стороны, встраиваемым в HTML и специально предназначенным для веб-приложений. Код PHP интерпретируется на веб-сервере, когда HTML-документ, в котором он встроен, запрашивается браузером. В результате интерпретации программного кода PHP обычно генерируется код HTML, который заменяет код PHP в документе HTML. Таким образом, веббраузер не видит кода PHP.

Язык Python

- Python (Лутц и Ашер, 2004) является относительно новым объектно-ориентированным и интерпретируемым языком скриптов.
- Используется для типов задач: системное администрирование, программирование CGI и другие относительно небольшие вычислительные задачи.
- Python является системой с открытым исходным кодом и доступен для наиболее популярных платформ.