

Лекция 12

Парадигмы программирования

Что такое парадигма программирования?

- Совокупность идей и понятий, определяющих стиль написания компьютерных программ (подход к программированию).
- Слово «парадигма» используется в программировании для определения семейства обозначений (нотаций), разделяющих общий способ (методику) реализаций программ. ([Диомидис Спинеллис](#))

Императивная и декларативная парадигмы

➤ Императивный подход:

- Последовательное выполнение инструкций
- Доступны данные после выполнения предыдущих инструкций
- Используется оператор присваивания/именованные переменные/подпрограммы

➤ Декларативный подход:

- Отсутствие последовательного выполнения инструкций
- Есть описание того, как поставлена задача и как должен выглядеть результат
- Не используется оператор присваивания/подпрограммы/циклы

Императивная и декларативная парадигмы

➤ Императивный подход:

- Pascal, Python, C, C++, Java

➤ Декларативный подход:

- SQL, Prolog

Один язык может сочетать в себе императивную парадигму и подвиды декларативной. Например, императивный Python поддерживает функциональную парадигму - подвид декларативной парадигмы.

Процедурное программирование

- Императивная парадигма
- Последовательное выполнение операторов
- Задача декомпозируется на шаги, которые решаются один за другим

Объектно-ориентированное программирование

- Программа - совокупность объектов
- Объект - экземпляр класса
- Класс - описание некоторого типа данных
- Классы образуют иерархию наследования

Prolog

- Prolog - логический (декларативный) язык программирования.
- Используется в области искусственного интеллекта, компьютерной лингвистики и нечислового программирования в целом.
- Логика программы выражается в терминах отношений, представленных в виде фактов и правил. Для того чтобы инициировать вычисления, выполняется специальный запрос к базе знаний, на которые система логического программирования генерирует ответы «истина» и «ложь».

Логическое программирование

domains

```
person = symbol(who_is_the_son :- parents(Son, Father, Mother), male(Son),  
               write("The son is ", Son, "."), nl.
```

predicates

male(person)

goal

female(person)

who_is_the_son

parents(person, person, person)

clauses

male("Frank").

male("Sam").

female("Mary").

female("Debbie").

parents("Sam", "Frank", "Mary").

parents("Debbie", "Frank", "Mary").

Результат работы программы

The son is Sam.

Функциональное программирование

- Функциональное программирование — раздел дискретной математики и парадигма программирования, в которой процесс вычисления трактуется как вычисление значений функций в математическом понимании последних (в отличие от функций как подпрограмм в процедурном программировании).
- Относится к декларативной парадигме.
- Не предполагается явное хранение состояния программы.

Функции в функциональном программировании

➤ Функция высшего порядка

Функция, которая может принимать на вход и/или возвращать другую функцию.

➤ Чистая функция

Функция зависит только от своих параметров и не взаимодействует с внешними данными. Это значит, что для одних и тех же данных гарантировано получится один и тот же результат (также говорят, что функция детерминирована и не имеет *побочных эффектов*).

Функциональное программирование

- Данные неизменяемые
- Программа - совокупность чистых функций
- Отсутствие циклов
- Использование функций высшего порядка
- Функция может быть сохранена в переменную
- Функция не зависит от имени, по которому мы к ней обращаемся.

Примеры ЯП

- Лисп
- F#
- Haskell

Источники и полезные ссылки

- Сравнение языков программирования по парадигмам:

https://ru.wikipedia.org/wiki/Сравнение_языков_программирования

- Пример программы на языке Пролог

<http://www.verim.org/project/prolog/listing/prezidenty>

- Википедия о языке Пролог:

[https://ru.wikipedia.org/wiki/Пролог_\(язык_программирования\)](https://ru.wikipedia.org/wiki/Пролог_(язык_программирования))

- Функциональное программирование:

https://ru.wikipedia.org/wiki/Функциональное_программирование