

Розвиток та суть функціонального програмування


Лекція 2


Передісторія ФП створена математиками

У 1920 р. радянський математик Моїсей Шонфінкель у праці "Про будівельні камені математичної логіки" проголосив нові тенденції в основах математики.

Суть нової концепції - обчислювальний (алгоритмічний) підхід до поняття функції, на відміну від традиційного її визначення як множини пар виду (аргумент, значення). У деякому змісті це було повернення до представлень математиків XVIII ст, але на новому рівні формалізму.

Шонфінкель розглядав функції як композиції базових "комбінаторів" і намагався знайти мінімальний набір таких комбінаторів, застосованих для визначення довільної функції.

- 
- Числення комбінаторів було незалежно розроблено і досліджено американським вченим Хаскелем Каррі і отримало назву "комбінаторної логіки".
 - В честь цього вченого було названо 2 мови функціонального програмування Carry і Haskell.




З'явилися інші формалізми, такі як
"лямбда-числення" Алонзо Чьорча, які
стали основою багатьох мов
функційного програмування і теорії
рекурсивних функцій.



Перша функціональна мова програмування

мова створена американським вченим
Джоном Маккарті - Lisp
(List Processing) у 1958 р.



Область застосування LISP - символні
обчислення, які є типовими для задач
штучного інтелекту.

Термін S-обчислення, S-вираз.
(Symbol)

Демонстрація принципів розв'язуваності проблем штучного інтелекту

1964 – Написана Дж.Вейценбаумом на LISP програма-співрозмовник "Еліза", що імітує мовну поведінку психоаналітика, дала позитивну відповідь на питання про можливість штучного розуму.



1978 - Поява LISP-комп'ютерів

Тобто LISP для таких ЕОМ була
машинною мовою

Стандартизація мови LISP (70-80-ті XX ст.)

Common Lisp – стандарт ANSI, найбільш відома спроба стандартизувати функційні мови. На жаль, мова виявилася надто складною. Її опис містить більше 1000 сторінок. Доступні як комерційні (Allegro, Harlequin, Corman), так і безплатні (GnuCL, CLISP, CMUCL) реалізації.

Діалекти

- Scheme — один з двох найбільш популярних в наші дні діалектів мови Lisp. Автори мови Scheme — Гай Стіл (Guy L. Steele) і Джеральд Сассмен (Gerald Jay Sussman) з Массачусетського Технологічного Інституту — створили його в середині 1970-х років.

- Erlang - мова розроблена компанією Ericsson, яка характеризується виразним синтаксисом і базується на співставленні зі зразком. Специфіка мови - орієнтація на паралельне програмування систем реального часу. Мова включає засоби для опису взаємодіючих процесів.


Сучасна функціональна мова

- F # - це функціональна мова програмування з сімейства мов . NET Framework, що підтримує функціональне програмування на додаток до імперативного (процедурного) і об'єктно-орієнтованому програмуванню. Мова розроблена Доном Сайма в Microsoft Research в Кембриджі, в даний час його розробку веде Microsoft Developer Division. F # достатньо тісно інтегрується з середовищем розробки Visual Studio і включений в постачання Visual Studio 2010; розроблені також компілятори для Mac і Linux.
- Деякі завдання вирішуються значно простіше і ясніше з використанням F # (у порівнянні з рішеннями на домінуючих ОО-мовах), особливо ті, що використовують математичний стиль програмування. Наприклад, написання компілятора вимагає роботи зі структурами даних і перетворень над ними.
- Microsoft інтегрувала середовище розробки F # в Visual Studio 2010. Компанія планує активно впроваджувати дану мову в розробку програмних систем, які самі з часом зможуть масштабуватися, наприклад, в залежності від кількості користувачів. Це непросто реалізувати в імперативних мовах програмування

Приклади використання LISP

- На Common Lisp написано конструктор електронних магазинів Yahoo! Store
- Редактор Emacs, написаний на Emacs-Lisp. Це один з найкращих редакторів для програмістів. Він «розуміє» синтаксис всіх популярних мов програмування (потужний текстовий редактор+компіляція).

- Примітивний діалект Lisp використовується в якості мови сценарієв музичного редактора Cakewalk.
- Найпростіший інтерпретатор Lisp використовується для опису імен та параметрів бібліотечних функцій у мові Perl


- 
- В якості мови текстового представлення даних різновид формату даних LISP використовується в ініціалізаційних файлах сервера Oracle і у файлах моделей Rational Rose.
 - Частково систему AutoCAD розроблено мовою Lisp.

■ Основне поняття


функціонального програмування
є функція (функціонал).

Математичні функції виражають зв'язок
між параметрами (входом) і
результатом (виходом) деякого
процесу.

Обчислення – це також процес, який має
вхід і вихід, функція є у повній мірі
адекватним засобом опису обчислень.



Функціональне програмування ставить своєю метою надання кожній програмі простої математичної інтерпретації. Ця інтерпретація повинна бути незалежна від деталей виконання і зрозуміла людям, які не мають особливих наукових пізнань у предметній області.



В основу мови покладено серйозний теоретичний апарат

- лямбда-обчислення Чьорча;
- алгебра спискових структур;
- теорія рекурсивних функцій.

Основні ідеї ФП

- Функціональна програма є набором визначень функцій.
- Функції визначаються через інші функції (правило композиції) або рекурсивно — через самих себе.
- У процесі виконання програми функції отримують параметри, обчислюють і повертають результат, у випадку необхідності обчислюються значення інших функцій.
- Програмуючи функційною мовою, програміст не повинен описувати порядок обчислень. Йому необхідно просто описати бажаний результат у вигляді системи функцій.

Властивості функціонального програмування

- 1) Представлення програми і даних відбувається однаково - через списки. Це дозволяє програмі обробляти інші програми і навіть саму себе.
- 2) Функційні мови, як правило, є інтерпретуючими мовами.
- 3) У функційних мовах символи не зв'язуються за замовчуванням з яким-небудь типом.
- 4) Функційні мови мають незвичний синтаксис із-за великої кількості дужок.
- 5) Функційні програми, написані для обробки символічних даних, є набагато коротші, ніж написані імперативними мовами.