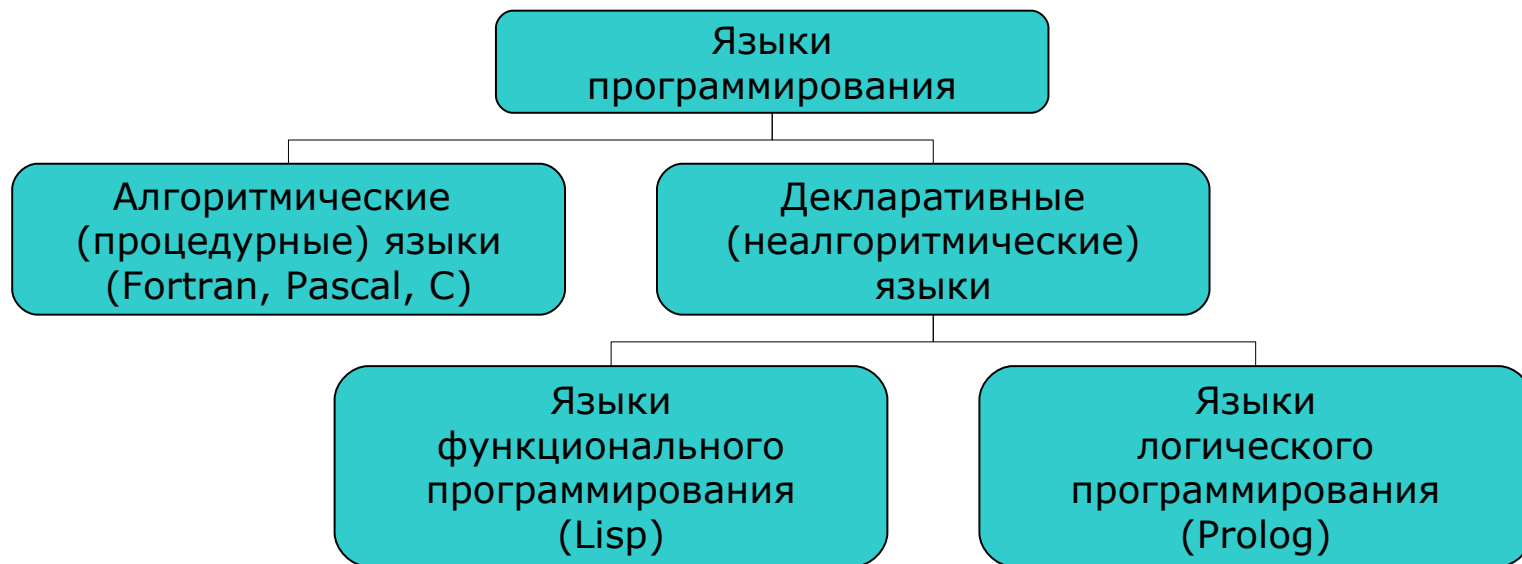


«Функциональное и логическое программирование»

Вводная лекция

Классификация языков программирования



В процедурных языках программа представляется как последовательность действий

В декларативных языках программа представляет собой множество отношений между некоторыми сущностями (объектами) решаемой задачи

Близость к естественному языку



Asm – ассемблер

АЯ – алгоритмические (процедурные) языки

ФП – языки функционального программирования

ЛП – языки логического программирования

ЕЯ – Естественные Языки (русский, английский и др.)

Язык функционального программирования Lisp

- Lisp – LISt Processing
- Разработчик – MIT
(Massachusetts Institute of Technology)
- Версии языка:
 - Common Lisp
 - XLisp



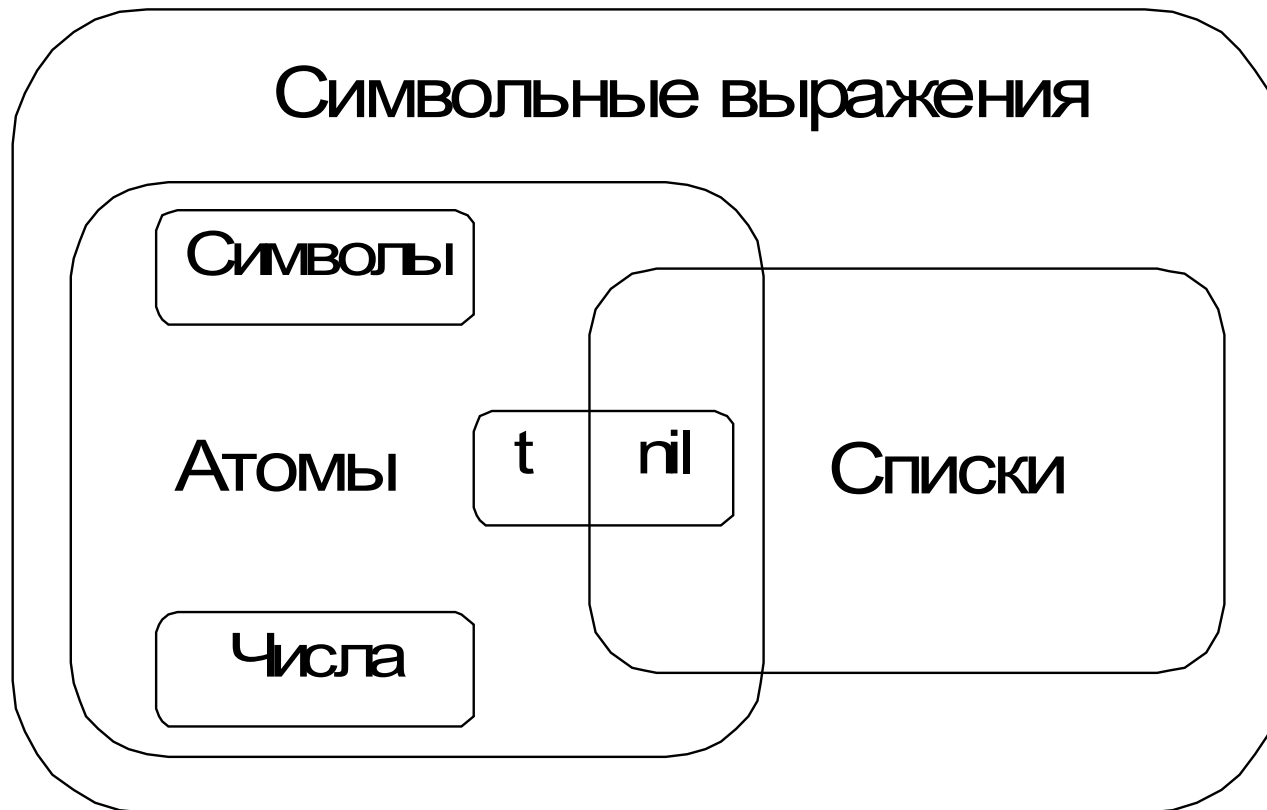
Основные отличия Lisp'a

- Одинаковая форма представления данных и программ – в виде списка
- Функциональный образ мышления
- Не требуется явное описание типов данных, используемых в программе
- Основной способ решения - рекурсия

Первая программа

- `> (+ 2 3)`
программа
данные
- `> (+ 2 3)`
5
- `> '(+ 2 3)`
(+ 2 3)
- `> (quote(+ 2 3))`
(+ 2 3)

Основы Lisp'a



Списки

- (2 a 5 str 2.5 34)
- Голова списка – первый элемент списка 2
- Хвост списка – все оставшиеся элементы списка, в свою очередь являющиеся самостоятельным списком (a 5 str 2.5 34)
- Пустой список – () или nil

Базовые функции

- `> (car '(1 2 3))`
1
- `> (cdr '(1 2 3))`
(2 3)
- `> (cons 1 '(2 3))`
(1 2 3)
- `> (atom '(1 2 3))`
NIL
- `> (equal '(1 2 3) '(1 2 3))`
T

Рекурсия

- > (defun factorial (n)
 (cond
 ((= n 0) 1)
 (t (* (factorial (- n 1)) n))
)
)
)
- >(factorial 3)
6

Язык логического программирования Prolog

- Prolog – PROgramming in LOGic
- Разработчик – PDC
(Prolog Development Center)
- Версии языка:
 - Turbo Prolog
 - PDC Prolog
 - Visual Prolog



Основные отличия Prolog'a

- Описание проблемы и правил ее решения
- Нахождение всех возможных решений с помощью механизма поиска с возвратом (backtracking)
- Простой синтаксис (атомарной структурой является предикат)



Области применения Prolog'a

- Разработка экспертных систем и оболочек экспертных систем
- Создание систем помощи принятия решений
- Разработка систем обработки естественного языка
- Построение планов действий роботов

Первая программа

- Волга – это река.
Ахтуба – это приток Волги.
Нечто является рекой при условии, что оно является притоком некоторой реки.
- река ("Волга").
река (X):– приток (X, Y).
приток ("Ахтуба", "Волга").

Первая программа

- Запрос
река (Z)
- Все возможные решения:
 - "Волга"
 - "Ахтуба"

Первая программа

- PREDICATES
river (symbol)
feeder (symbol, symbol)

- CLAUSES

river ("Волга"). ————— | Факт

river (X):– feeder (X, Y).

feeder ("Ахтуба", "Волга"). ————— | Факт

- Goal: river (Z) ————— | Вопрос

- Результат: Yes

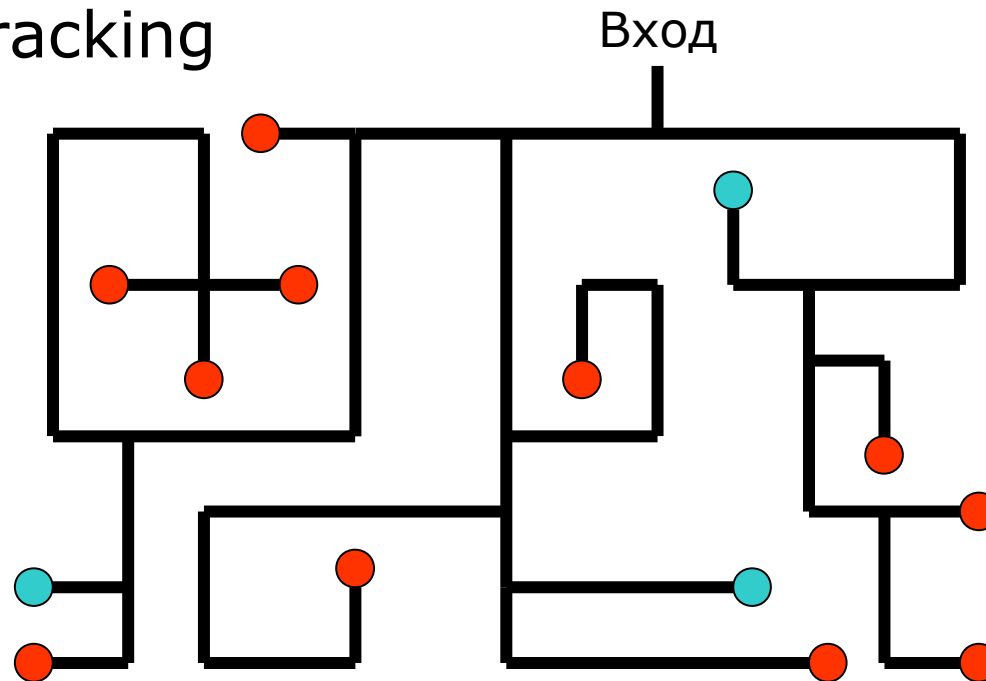
Z="Волга"

Z="Ахтуба"

Правило
вывода

Основные способы решения

- Поиск с возвратом
backtracking



- Рекурсия

Поиск с возвратом

- Недоказательство цели
- Откат к цели, которую можно передоказать (недетерминированной цели)

- PREDICATES

male (integer) ; мужчина

female (integer) ; женщина

start ; формирование пар

CLAUSES

male ("Иван"). male ("Юрий"). female ("Оля"). female ("Катя").

start:- male (X), female (Y), write (X, Y), fail.

start.

GOAL

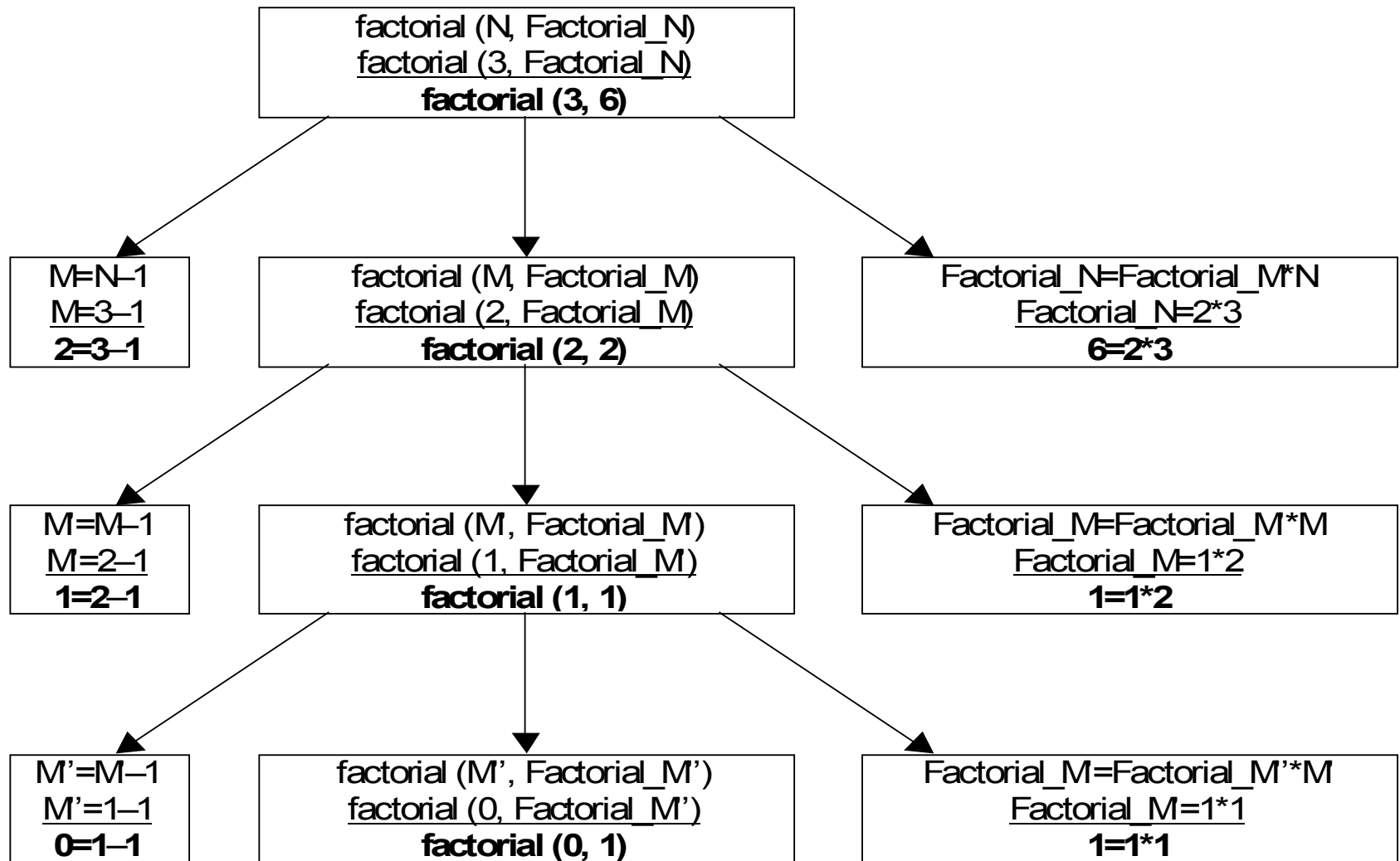
start.

- Результат: "ИванОля", "ИванКатя", "ЮрийОля", "ЮрийКатя"

Рекурсия

- PREDICATES
factorial (integer, real)
- CLAUSES
factorial (0, 1).
factorial (N, FactN):- M=N-1,
 factorial (M, FactM),
 FactN= FactM*N.
- GOAL
factorial (3, FactN), write ("FactN=", FactN).

Дерево целей



Рекурсивные типы данных

- Списки [1, 2, 3]
- Деревья (бинарные, упорядоченные)

