

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Кузбасский государственный технический университет имени Т. Ф. Горбачева»

Кафедра информационных и автоматизированных
производственных систем

Составитель
А. В. Протоdjяконов

Арифметические вычисления и сравнения в Прологе

Методические указания к лабораторным работам по дисциплине
«Системы искусственного интеллекта» для студентов
направления подготовки 15.03.04 «Автоматизация
технологических процессов и производств», профиль
«Автоматизация технологических процессов и производств»,
очной формы обучения

Рекомендовано учебно-методической комиссией направления
15.03.04 «Автоматизация технологических процессов
и производств» в качестве электронного издания
для использования в учебном процессе

КЕМЕРОВО 2017

Рецензенты:

И. В. Чичерин – доцент, кандидат технических наук, председатель учебно-методической комиссии направления 15.03.04 «Автоматизация технологических процессов и производств»

А. Н. Трусов – доцент кафедры информационных и автоматизированных производственных систем

Протоdjаконов Андрей Владимирович

Арифметические вычисления и сравнения в Прологе: методические указания к лабораторным работам по дисциплине «Системы искусственного интеллекта» [Электронный ресурс]: для студентов направления подготовки 15.03.04 «Автоматизация технологических процессов и производств», профиль «Автоматизация технологических процессов и производств», очной формы обучения / сост. А. В. Протоdjаконов; КузГТУ. – Электрон. дан. – Кемерово, 2017. – Систем. требования: Pentium IV; ОЗУ 8 Мб; Windows 95; мышь. – Загл. с экрана.

Составлено в соответствии с рабочей программой дисциплины «Системы искусственного интеллекта» и предназначено для проведения лабораторных занятий при изучении арифметических вычислений и сравнений в Прологе.

© КузГТУ, 2017

© Протоdjаконов А. В.,
составление, 2017

1. Цель работы

Цель – изучить возможности встроенных предикатов и функций для выполнения вычислений, сравнений и операций ввода-вывода.

2. Арифметические вычисления и сравнения

2.1. Функции и предикаты

Возможности вычислений в Visual Prolog аналогичны соответствующим возможностям таких языков программирования, как C, Basic, Pascal.

Visual Prolog имеет полный набор встроенных математических функций и предикатов, которые используют целые и вещественные значения. Их список приведен в таблице 1.

Таблица 1. Арифметические функции и предикаты

$X \bmod Y$	Возвращает остаток от деления X на Y .
$X \operatorname{div} Y$	Возвращает частное от деления X на Y .
$\operatorname{abs}(X)$	Модуль X .
$\cos(X)$, $\sin(X)$, $\tan(X)$	Косинус, синус, тангенс X соответственно (X в радианах).
$\operatorname{arctan}(X)$	Арктангенс X .
$\exp(X)$	Возводит e в степень X
$\ln(X)$, $\log(X)$	Натуральный и десятичный логарифмы X .
$\operatorname{sqrt}(X)$	Квадратный корень X .
$\operatorname{random}(X)$	Присваивает X случайное число ($0 \leq X < 1$).
$\operatorname{random}(X, Y)$	Присваивает Y случайное целое число ($0 \leq Y < X$).
$\operatorname{randominit}(\operatorname{SEED})$	Инициализирует генератор случайных чисел. При постоянном SEED предикат $\operatorname{random}()$ будет выдавать одно и то же значение (используется для статистического тестирования).
$\operatorname{round}(X)$	Округляет X .
$\operatorname{trunc}(X)$	Усекает (отбрасывает дробную часть) X .
$\operatorname{val}(\operatorname{domain}, x)$	Явное преобразование числовых доменов (domain – имя возвращаемого домена).

Для возведения числа в произвольную степень X можно использовать комбинацию функций \exp и \ln . Например,

$$12^{8,3} = \exp(8,3 * \ln(12)) \quad \text{или} \quad x^Y = \exp(Y * \ln(x))$$

Тригонометрические функции требуют, чтобы X был величиной, представляющей угол в радианах.

В арифметических выражениях в Visual Prolog используются: бинарные операторы (+, -, *, /), унарные операторы (-, +); скобки.

Приоритеты операций в порядке возрастания: бинарные операторы (-, +), бинарные операторы (/ , *, mod, div), унарные операторы (-, +), скобки.

2.2. Сравнение

В Visual Prolog используются следующие операторы сравнения: <, <=, =, >=, >, <> или ><.

Сравнивать можно как числовые, так и символьные выражения.

При сравнении символов Пролог использует значение кодов ASCII для каждого символа. Например, сравнение 'a' < 'b' преобразуется в арифметическое выражение 97 < 98.

Когда сравниваются строки или идентификаторы, результат зависит от сравнения символов на соответствующих позициях. Результат равен тому, который получился бы при сравнении первых символов в том случае, если эти два символа не равны. Если же они равны, то Пролог сравнивает следующую пару символов и возвращает результат тогда, когда они не равны. В противном случае проверяется следующая пара и т.д. Если конец достигнут, и пара различных символов не найдена, более короткая строка считается меньшей.

Сравнение "antony" > "antonia" оценивается как истинное, т.к. первая пара различных символов у и і имеют значения 79 и 69 в кодах ASCII. Аналогично, сравнение "aa" > "a" - истинно.

Идентификаторы не могут сравниваться непосредственно из-за синтаксиса. Например, при сравнении P1=peter, P2=sasha идентификатор peter не может непосредственно сравниваться с

идентификатором *sasha*. Они должны быть связаны с переменными, которые сравниваются или записаны как строки: $P1 > P2$.

2.3. Генератор случайных чисел

Для генерации случайных чисел в Visual Prolog предусмотрены два стандартных предиката. Один возвращает случайное вещественное число в диапазоне от 0 до 1, другой возвращает случайное число в диапазоне от 0 до данного числа.

Предикат **random/1** возвращает случайное вещественное число `RandomReal`, удовлетворяющее условию: $0 \leq \text{RandomReal} < 1$. Формат предиката: `random (RandomReal)`.

Предикат **random/2** имеет два аргумента, его формат: `random(MaxValue, RandomInt)`. Этот предикат ставит в соответствие `RandomInt` случайное число, удовлетворяющее условию: $0 \leq \text{RandomInt} < \text{MaxValue}$. Предикат `random/2` работает быстрее, чем **random/1**, т.к. имеет целочисленную арифметику.

Предикат **randominit/1** имеет формат: `randominit(Seed)`. По умолчанию случайное начальное значение генерируется как функция системного времени, а аргумент `Seed` устанавливает начальное значение. Основное назначение `randominit` – предоставить повторяемую последовательность псевдослучайных чисел для статистического тестирования.

Целочисленная и вещественная версии `random` используют одинаковое начальное значение и базовый генератор чисел.

3. Предикаты ввода-вывода

Вывод на экран. Для вывода на экран используется предикат **write()**. Его синтаксис:

1) **write**(*список переменных для вывода [управляющие символы]*). Переменные разделяются запятыми;

2) **write**(“*строка для вывода [управляющие символы]*”).
Управляющие символы – `\t`, `\r`, `\n` и т.д.,

где `\t` – табуляция;

`\r` – возврат каретки;

`\n` – начало новой строки.

Для того чтобы начать вывод в новую строку используется предикат **nl** – *new line*.

Пример:

PREDICATES

s(string)

CLAUSES

s("Значение").

GOAL

s(W),

write("\tПрограмма пример\nВторая строка\n"),

write("Ответ =", "\t", W), nl,

write("Конец программы"), nl, nl.

После запуска программа выдает следующий результат:

```

    Программа пример
    Вторая строка
    Ответ =  Значение
    Конец программы

```

W=Значение

1 Solution

Последние две строки программа, а точнее утилита Test Goal, выдает автоматически.

Ввод с клавиатуры. Для ввода данных с клавиатуры используются предикаты:

readint(переменная) – ввод одной переменной типа INTEGER;

readreal(переменная) – ввод одной переменной типа REAL;

readchar(переменная) – ввод одной переменной типа CHAR.

Во всех случаях, кроме последнего, ввод необходимо заканчивать нажатием клавиши ENTER.

4. Порядок выполнения работы

1. Составить программу вычисления арифметического выражения.

Варианты заданий приведены в таблице 2.

При создании проекта в окне **Application Expert** на вкладке **Target** установить значение **UI Strategy** в **Easywin**. После создания проекта необходимо установить требуемые опции компилятора. Для чего открыть пункт меню **Options** → **Project** → **Compiler Options**. В появившемся окне выбрать пункт **Nondeterm**, снять флажки с пунктов **Non Quoted Symbols**, **Strong Type Conversion Check** и **Check Type of Predicates**.

Структура программы включает разделы предложений, предикатов и целей:

Predicates

/*описываются предикаты (аналоги прототипов функций на C++) с используемыми доменами.*/

```
result(real, real)
result2(real, real, real, real, real)
```

Clauses

/*описание правил (аналог описания функций C++). Правило – это конструкция Пролога, которая описывает, что можно логически вывести из других данных. Правило – это свойство или отношение, которое достоверно, когда известно, что ряд других отношений достоверен. Синтаксически эти отношения разделены запятыми.*/

```
% Например, вычисление суммы 2-х чисел Var1 и Var2
result(Var1, Var2):-
    Var=Var1+Var2,
    write(Var), nl.
```

```
% Вычисление значения выражения A*B/C+D-E
result2(A, B, C, D, E):-
```

```
F = A*B/C+D-E,
write(F), nl.
```

Goal

/* описываются цели. При запуске Пролог автоматически выполняет цель*/

```
readreal(X) % ввод переменной X,
result(X, 2),
result2(5, 4, 3, 2, 1).
```

2. Для каждого задания написать 3 варианта программ:

- с заданием переменных в теле программы;
- с введением данных с клавиатуры;
- с заданием значений переменных с помощью генератора случайных чисел.

3. Используя тригонометрические функции Visual Prolog построить таблицу синусов, косинусов, тангенсов на экране. Левая колонка таблицы должна содержать величину угла в градусах, начиная от 0 до 360 градусов с шагом 60 градусов. Таблица должна иметь название, названия колонок, границы (линии разделяющие колонки и строки). Для формирования границ использовать знаки тире, подчеркивания и вертикальной линии (“-”, “_”, “|”). В результате должно получиться примерно следующее:

```
Программа пример
-----
|  Угол  |      Sin      |
-----
|  60    | 0.8660253949 |
-----
```

Конец программы

```
Otvet=0.8660253949
1 Solution
```

4. Написать программу: В компании из десяти человек выбрать трех, которые побегут за Клиным, чипсами, раками. Определить счастливого обладателя приза BMW.

5. Варианты задания

Таблица 2. Варианты заданий

B1	1. $(x^2 + 6^Y) / (8 - e^2)$ 2. $Y \cdot \ln(x) \cdot (\cos(60^\circ) / \operatorname{tg}(45^\circ)) + Y$
B2	1. $(100^2 / (e^X + 6^Y + 8 \cdot X \cdot Y))^{1/2}$ 2. $\lg(\sin(45^\circ) / e^3) - X^Y$
B3	1. $e^{(X^2 - Y^2) / \lg 5} + \ln(X + Y)$ 2. $(X \cdot \operatorname{tg}(10^\circ) + Y \cdot \cos(85^\circ))^{1/2}$
B4	1. $(\lg(100 \cdot X) - 8 \cdot Y) / e^{\ln 2}$ 2. $(X \cdot \operatorname{arctg}(1) / \lg(\ln 2) / Y)^{1/2}$
B5	1. $e^{(\ln 4 + \lg 20)} + X^3 - 5 \cdot Y$ 2. $((8^X - Y^2) \cdot (\cos 60^\circ)^{1/2}) / \sin 30^\circ$
B6	1. $2 \cdot \ln(x) - 1/x + 0.5 \cdot x^2$ 2. $Y^{(2 \cdot X)} \cdot X + \ln(10 \cdot Y)$
B7	1. $(X + 6)^2 + Y / \lg(3.56)$ 2. $(25 + Y + X^2) / (6Y^X \cdot \sin(330^\circ))$
B8	1. $X \cdot 2 + 56 - Y / X + \cos(X \cdot \pi)$ 2. $(\ln(27))^{X \cdot Y} + 9(X - Y)$
B9	1. $(X + 9 \cdot \ln(Y) - X^Y) / 35$ 2. $1 + 3XY^2 - \ln(6 \cdot Y)$
B10	1. $Y + X^4 - \ln(14X)$ 2. $(9 + 4X) / Y + 7X^2 - e^{\cos(3X)}$
B11	1. $25 + Y / 8 + X^2 + 6Y^X - 9 \cdot \ln(X)$ 2. $y^e - x^{-4} + \lg(27 + y)$
B12	1. $4X + 6^Y - 9X^3$ 2. $\operatorname{COS}(12 \cdot Y) + Y^X + (\ln(125)) / (3 + x)$
B13	1. $6^Y - 9X + \ln(9Y) - 36$ 2. $(\lg(27))^2 + 9 + 2X - Y \cdot X$
B14	1. $\ln(14X) - (\lg(27Y))^2 + 9X / Y$ 2. $\operatorname{SIN}(3Y) + \operatorname{COS}(X^{3Y})$
B15	1. $\operatorname{tg}(3X) + \ln(9Y) + (X^{2Y} \cdot Y^{2X})^{0.5}$ 2. $\ln(X + Y) + (\lg(X^4 + 3Y)) + X^9$
B16	1. $\operatorname{COS}(3.8 \cdot Y) + \lg(6.5X) - 12XY^6$ 2. $X^4 - \ln(14) + 3X^2Y + 1/9$
B17	1. $\operatorname{COS}(Y + 2X) + 6^Y - 9X - 4/35$ 2. $\ln(Y + 5) + 6^{Y(X+3)} + 24Y$

B18	1. $e^{(\sin(3X)+1/2\cos(XY+2))}+10*\ln(31)$ 2. $\ln(9Y) +\cos(XY+1)-2*X^Y$
B19	1. $12Y - (X^2+Y^2)^{0.5} + \ln(7X)$ 2. $3X+5YX-125Y^{-4}+e^X$
B20	1. $1/5X^3+10Y^2-5/13XY+X^{\cos(2\pi X)}$ 2. $\ln(3+(\sin(3X))^4+(\cos(2Y))^2)+e^{10X}$

6. Контрольные вопросы

1. Какие выражения можно сравнивать в Прологе?
2. Какие встроенные предикаты используются для ввода и вывода?
3. Каково основное назначение предиката `randominit`?
4. Как происходит сравнение строковых переменных?
5. Чем отличаются бинарные операторы от унарных?
6. В чем состоит отличие встроенных предикатов **random/1** и **random/2**?

Литература

1. Адаменко А. Н. Логическое программирование и Visual Prolog / А. Н. Адаменко, А. М. Кучуков. – СПб.: БХВ-Петербург, 2003.
2. <http://wikiru.visual-prolog.com/>
3. Цуканова Н. И. Логическое программирование на языке Visual Prolog: учеб. пособие для вузов / Н. И. Цуканова, Т. А. Дмитриева. – М.: Горячая линия – Телеком, 2008 – 144 с.