Міністерство освіти і науки України

Львівський національний університет імені Івана Франка

Факультет електроніки та комп'ютерних технологій

Кафедра радіоелектронних та комп'ютерних систем

Звіт

про виконання лабораторної роботи №1

**Cередовище Visual Prolog: основні поняття, інтерфейс**

Виконав студент

групи Феі- 54

Вегера Северин

Викладач:

доц. Бабич О. Й.

Львів — 2018

Короткі теоретичні відомості:

**Проло́г** — мова логічного програмування загального призначення, пов'язана зі штучним інтелектом та математичною лінгвістикою.

Пролог має корені в математичній логіці, та, на відміну від багатьох інших мов програмування, є декларативною: логіка програми виражається в термінах відношень, представлених як факти та правила. Обчислення ініціюється запуском запиту над цими відношеннями.

Цю мову програмування спочатку було задумано групою навколо Алана Кольмерое у Марселі на початку 1970-тих, а першу систему Пролог було розроблено у 1972-му Аланом Кольмерое та Філіпом Русселем.

Пролог була однією з перших логічних мов програмування, й залишається найпопулярнішою серед таких мов і на сьогодні, маючи багато безкоштовних та комерційних реалізацій. Її застосовували як для доведення теорем, експертних систем, так і для її початкової області призначення, обробки природної мови. Сучасні середовища Прологу підтримують як створення графічних інтерфейсів користувача, так і адміністративні або мережеві застосування.

**Пролог** створювався, в першу чергу, як спеціалізована мова, призначена для вирішення завдань, що відносяться до галузі штучного інтелекту. Саме в цій області повною мірою виявляються всі її достоїнства і переваги перед мовами широкого призначення. **Пролог-програма** являє собою набір логічних описів, що визначають мету, заради досягнення якої вона написана. Послідовність вирішення закладена в математичній моделі мови, яка реалізується в її внутрішньому механізмі і забезпечує пошук результату на основі **вихідного логічного опису**. Мова має суворе математичне обґрунтування і орієнтована на використання концепцій і методів логіки **предикатів першого порядку**. **Предикат** - це логічна функція від n аргументів, що має тільки два значення "істина" і "не істина". Аргументи предиката - це пов'язані відношенням **об'єкти**. Синтаксис предиката:

**<імя предиката> (<аргумент 1>, <аргумент2>…<аргумент п>)**

На cьогоднішній день Пролог є живою мовою, яка активно розвивається, хоча і не має такого широкого розповсюдження як C++, С# і створені на їх основі інтегровані середовища розробки. З початку своєї появи в 70-х рр. ХХ-го століття система програмування на Пролозі існувала як **інтерпретатор**. У 1988 році з'явилася система Turbo Prolog, а потім і Visual Prolog, що включають **компілятор** і **інтегроване середовище** розробки програм. Розроблена не так давно 7-ма версія Visual Prolog є **об'єктно-орієнтованим середовищем** розробки логічних програм. Математична основа і механізм роботи Прологу однакові для інтерпретатора і компілятора; разом з тим у побудові програм є деякі суттєві відмінності. Класичні праці, що пояснюють механізм роботи і області застосування Прологу, написані для інтерпретатора. Тому, незважаючи на те, що компілятори витіснили інтерпретуючі системи ми будемо розбирати деякі питання програмування як для компіляторів, так і для інтерпретаторів. При цьому система програмування з **інтерпретатором** називається **традиційною системою Пролог**, а **з компілятором - Visual Prolog (VIP)**. Якщо мова йде про загальні питання, то називається просто Пролог.

**Інтерпретатор** мови програмування (interpreter) — програма чи технічні засоби, необхідні для виконання інших програм, вид транслятора, який здійснює пооператорну (покомандну, полінійчату) обробку, перетворення у машинні коди та виконання програми або запиту (на відміну від компілятора, який транслює у машинні коди всю програму без її виконання).

Постановка задачі:

Є база даних, що містить наступні факти:

родич("Ілля", "Марина") - батько

родич("Марина", "Іра"). - мама

родич("Олена", "Іван"). - мама

родич("Микола", "Іра"). - батько

родич("Ольга", "Олексій"). - мама

родич("Марина", "Олександр"). - мама

родич("Сергій", "Іван"). - батько

Завдання:

1) чи вірно, що Марина є мамою Іри?;

1.1) хто є мамою Іри

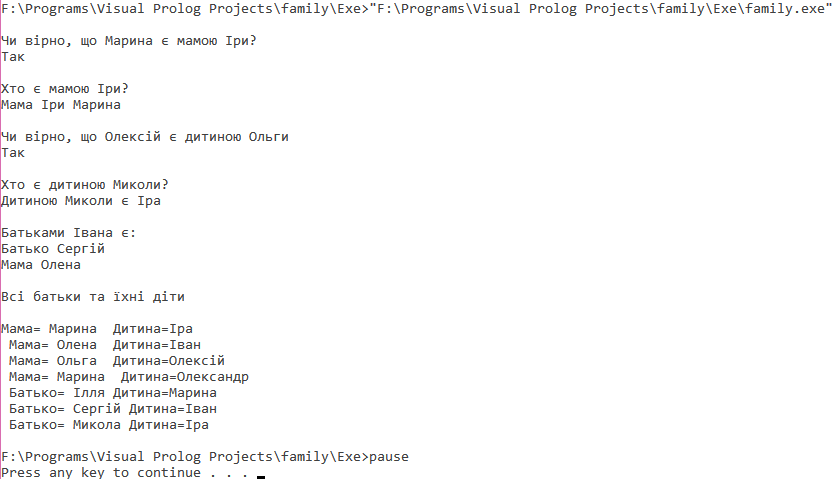
2) чи вірно, що Олексій є родичем Ольги;

3) хто є дитиною Миколая;

4) хто батьки Івана;

5) всіх батьків і їхніх дітей.

**Результати:**



**Остаточний код задачі:**

% Copyright

implement main

open core

domains

name = string.

class predicates

батько : (name, name) nondeterm anyflow.

мама : (name, name) nondeterm anyflow.

clauses

батько ("Ілля", "Марина").

батько ("Сергій", "Іван").

батько ("Микола", "Іра").

мама ("Марина", "Іра").

мама ("Олена", "Іван").

мама ("Ольга", "Олексій").

мама ("Марина", "Олександр").

clauses

%1

run() :-

console::init(),

stdio::write(""), stdio::nl,

stdio::write("Чи вірно, що Марина є мамою Іри?"), stdio::nl,

if (мама ("Марина", "Іра")) then

stdio::write("Так"), stdio::nl

else

stdio::write("НІ") , stdio::nl

end if,

fail.

%1.1

run() :-

console::init(),

stdio::write(""), stdio::nl,

stdio::writef("Хто є мамою Іри?"), stdio::nl,

мама (X,"Іра"), stdIO::writef("Мама Iри %\n",X),

fail.

%2

run() :-

console::init(),

stdio::write(""), stdio::nl,

stdio::write("Чи вірно, що Олексій є дитиною Ольги"), stdio::nl,

if (мама ("Ольга", "Олексій")) then

stdio::write("Так"), stdio::nl

else

stdio::write("НІ") , stdio::nl

end if,

fail.

%3

run() :-

console::init(),

stdio::write(""), stdio::nl,

stdio::write("Хто є дитиною Миколи?"), stdio::nl,

батько ("Микола",X), stdIO::writef("Дитиною Миколи є %\n",X),

fail.

%4

run() :-

console::init(),

stdio::write(""), stdio::nl,

stdio::write("Батьками Івана є:"), stdio::nl,

батько (X,"Іван"), stdIO::writef("Батько %\n",X),

мама (Y,"Іван"), stdIO::writef("Мама %\n",Y),

fail.

%5

run() :-

console::init(),

stdio::write("\nВсі батьки та їхні діти\n"), stdio::nl,

мама(X,Z), stdIO::writef("Мама= % Дитина=%\n ",X,Z),

fail.

run() :-

console::init(),

батько(Y,Z), stdIO::writef("Батько= % Дитина=%\n ",Y,Z),

fail.

run() :-

console::init(),

stdIO::writef("").

end implement main

goal

console::runUtf8(main::run).

**Висновок:**

На цій лабораторній роботі, я вивчив основні поняття та інтерфейс середовища Visual Prolog. Ознайомився із основними термінами і структурою програми Пролог. Реалізував код, який виконує необхідні запити.