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 №6

" Арифметичні обчислення"

Виконав студент

групи Феі- 54

Вегера Северин

Викладач:

доц. Бабич О. Й.

Львів — 2018

Короткі теоретичні відомості:

Всі твердження програми на Пролозі діляться на **факти, правила і питання**. Факти відображають поточний стан предметної області, містять конкретну інформацію і є істинними предикатами. **Факти** відповідають простим безумовним висловлюванням.

**Факти**, які приведені всередині програми вводять одночасно **об'єкти** та їх класифікацію (**відношення**). Наприклад, факт **закуска ("салат з яйцем")** показує, що салат з яйцем є закускою.

Синтаксично правильно записаний факт має наступну структуру предиката (рис. 1).

На рис. 1 у списку аргументів перераховуються імена об'єктів (не більше 255), пов'язаних даними відношенням. Аргументи в списку відділяються один від одного комами, в деяких випадках можуть бути відсутні. Якщо аргумент являє собою ім'я конкретного об'єкта (властивості) або число, то він є **константою** Прологу.

ФАКТ <імя вітношення> (<список аргументів>).

Рис. 1. Структура предиката факт

Визначаючи за допомогою фактів відношення між об'єктами, необхідно враховувати порядок, в якому перераховуються їх імена всередині круглих дужок. Вибравши один раз який-небудь порядок, ви повинні всюди слідувати йому і далі. Наприклад, факт **батьком ("Петро", "Іван")** означає, що Петро є батьком Івану, а факт **батьком ("Іван","Петро")** говорить вже зовсім про інше, а саме, що Іван є батьком Петрові. Одне і те ж твердження, записане у вигляді факта, може по-різному інтерпретуватися. Тільки автор програми визначає справжню інтерпретацію імен об'єктів i порядок проходження аргументів, і їм він повинен слідувати в процесі написання всієї программи, відображаючи в коментарях сенс записаних ним висловлювань. **Коментар** - це текст, укладений між символами / \* ... \* / або рядок, що починається зі знака %.

Завдання:

**1. Обчислити значення виразу Z = (2 \* X + Y) / (X-Y) для введених X і Y.**

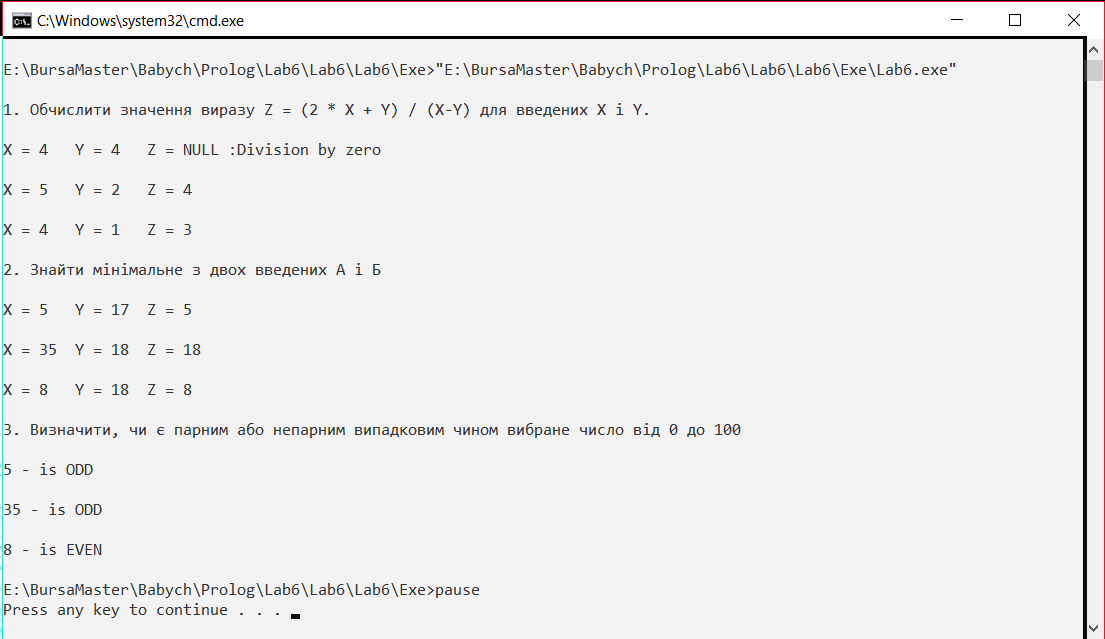
**2. Знайти мінімальне з двох введених А і Б.**

**3. Визначити, чи є парним або непарним випадковим чином вибране число від 0 до 100.**

Остаточний код задачі:

implement main  
    open core  
constants  
className = 'main  
class predicates  
result : (real, real, real) procedure (i, i,o).  
min : (integer, integer, integer) procedure (i, i,o).  
evenOrOdd : (integer) procedure (i).  
  
clauses  
result(X,Y,Z):-  
    stdIO::writef("\nX = %\tY = %", X, Y),  
    if (X = Y), !  
        then Z = 0,  stdIO::writef("\tZ = NULL :Division by zero\n")  
        else  Z = (2 \* X + Y) / (X - Y),  
        stdIO::writef("\tZ = %\n", Z)  
    end if.  
  
min(X,Y,Z):-  
    stdIO::writef("\nX = %\tY = %", X, Y),  
    if (X <= Y), !  
        then  Z = X,  
        stdIO::writef("\tZ = %\n", Z)  
        else  Z = Y,  
        stdIO::writef("\tZ = %\n", Z)  
    end if.  
  
evenOrOdd(X):-  
    stdIO::writef("\n%", X),  
    if (X mod 2 = 0), !  
        then  
        stdIO::writef(" - is EVEN\n")  
        else  
        stdIO::writef(" - is ODD\n")  
    end if.  
clauses  
  
 run() :-  
    console::init(),  
    stdIO::writef("\n1. Обчислити значення виразу Z = (2 \* X + Y) / (X-Y) для введених X і Y.\n"),  
   result(4, 4, Z1),  
   result(5, 2, Z2),  
   result(4, 1, Z3),  
    fail.  
  
    run() :-  
    console::init(),  
    stdIO::writef("\n2. Знайти мінімальне з двох введених А і Б\n"),  
   min(5, 17, Z1),  
   min(35, 18, Z2),  
   min(8, 18, Z3),  
    fail.  
  
    run() :-  
    console::init(),  
    stdIO::writef("\n3. Визначити, чи є парним або непарним випадковим чином вибране число від 0 до 100\n"),  
   evenOrOdd(5),  
   evenOrOdd(35),  
   evenOrOdd(8),  
    succeed.  
  
end implement main  
  
goal  
    console::runUtf8(main::run).

Результати:



Висновок:

На цій лабораторній роботі, я ознайомився із арифметичними операціями середовища Visual Prolog і створив проект, який розв’язує поставлені завдання. Склав правила для обчислення. Реалізував код, який виконує необхідні запити.