

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ



**Дніпровський національний університет  
залізничного транспорту імені академіка В. Лазаряна**

Кафедра «Комп'ютерні інформаційні технології»

**Лабораторна робота №1**

**з дисципліни «Конструювання програмного забезпечення»**

**на тему: «Вивчення системи SWI-PROLOG та методів розробки логічних програм»**

Виконав:  
студент гр.ПЗ1911  
Сафонов Д. Є.  
Прийняв:  
Іванов О.П.

Дніпро, 2021

**Тема.** Вивчення системи SWI-PROLOG та методів розробки логічних програм.

**Завдання.** Створити програму на основі фактів (кількість фактів 10-20 штук), яка оброблює тривимірні матриці(варіант 3).

**Текст програми.** [github](#)

## **Основні розділи структури програми**

Програма в пролозі складається з фактів - набору даних, аксіом; правил - нових даних, які отримуються з сукупності фактів; і мети - перевірки якихось правил на правдивість та/або знаходження деяких змінних. Також користувач може задавати додаткові цілі після запуску програми в інтерпретаторі, вони називаються запитами.

## **Результати виконання програми**

### **Висновки**

Мова пролог не відноситься до імперативних мов програмування, вона є декларативною мовою, а саме — логічною. Ця мова дуже відрізняється від імперативних мов тим, що вона описує результат, який потребується отримати після виконання програми, а не алгоритм із допомогою, якого цей результат треба шукати.

Програма складається із фактів, правил, та запитів. Факти складаються з атомів і показують наявність деяких відношень між цими атомами. Правила можуть співвідносити існуючі факти та виводити нові. Запити у свою чергу є фінальними цілями програми. Атоми можуть бути одним словом з маленької літери, або іншого вигляду, якщо закрити їх у лапки. У пролозі не існує циклів, тож будь які ітеративні алгоритми мають бути описані у рекурсивному вигляді. Окрім атомів існують й інші типи даних, а саме: числа(цілі та з плаваючою точкою), змінні(використовуються в правилах та запитах схожим чином як аргументи у імперативних мовах) та складені терми, які складаються з функтора(атома), та інших термів(об'єктів типів даних або складені терми), також існують списки(набір впорядкованих термів) та стрічки(набір символів у подвійних лапках).

Перевагою декларативних мов програмування є розмір їх програм, що дозволяє програмістам легше їх доповнювати, та допускати менше помилок.