

Міністерство освіти і науки України
Дніпровський національний університет імені Олеся Гончара
Факультет прикладної математики і комп'ютерних технологій
Кафедра комп'ютерних технологій

ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ № 1
з курсу «Моделі та методи штучного інтелекту»
на тему «Розробка предиката»
Варіант №14

Виконав:
студент гр. ПА-22-2
Овдієнко Андрій

Дніпро
2025

Зміст

1. Постановка задачі	3
2. Опис розв'язку.....	4
3. Вихідний текст програми розв'язку задачі.....	6
(основні фрагменти з коментарями)	6
4. Опис інтерфейсу програми (керівництво користувача)	8
5. Опис тестових прикладів	9
6. Виправлення помилок	10
 6.1 Хід усунення помилок	10
 6.2 Код програми	10
 6.3 Прикла роботи програми	12

1. Постановка задачі

Використовуючи предикати `parent(string parent, string son-daughter)`, `women(symbol)` і `man(symbol)`, розробити предикат, що визначає таке поняття: Свояк.

Написати програму для пошуку і видачі на екран відповідного родича в базі даних.

2. Опис розв'язку

Спочатку потрібно зрозуміти, хто такий свояк. Простіше кажучи, свояк – це чоловік сестри дружини. Тобто, нехай будуть чоловік і дружина. У дружини є сестра, а чоловік цієї сестри – свояк.

Тепер перейдемо до середовища розробки. Слід використовувати мову Prolog. Костянтин Євгенович Золотко рекомендував встановити Visual Prolog 5.2. Встановимо його з офіційного сайту [visual-prolog](https://visual-prolog.software.informer.com/download/) (Рисунок 1).

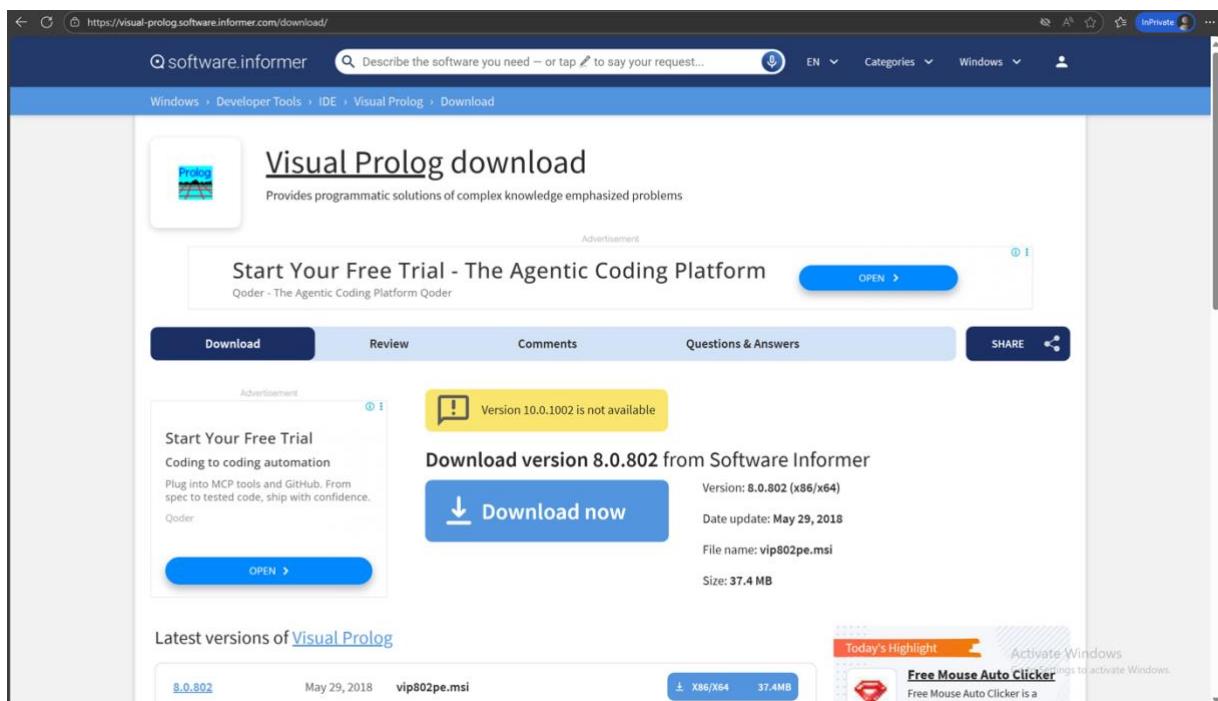


Рисунок 1 – Офіційний сайт для завантаження Visual Prolog 5.2.

Найголовніше – почати розуміти, який блок для чого використовується. DOMAINS (domains) – розділ, де оголошуються домени та їх типи. Щоб мінімізувати предикати – ми оголошуємо один предикат – person, і надамо symbol типу.

PREDICATES (predicates) – розділ, призначений для оголошення предикатів та доменів аргументів предикатів. У постановці задачі ми запропонували три предикати: parent(string parent, string son-daughter) – перший аргумент – це один з батьків, а другий – син або дочка; women(symbol) – один аргумент, і це жінка; man(symbol) – один аргумент, і це чоловік – ми будемо використовувати ці три предикати як факти, тобто наповнювати ними базу даних.

Для зручності додамо чотири правила: *husband(person, person)* – перша особа – це чоловік іншої особи (мається на увазі жінки, звичайно); *woman(person, person)* – перша особа – це дружина іншої особи (мається на увазі чоловіки, звичайно); *sister(person, person)* – перша особа є сестрою іншої особи (не має значення, чоловік це чи жінка); *brother_in_law(person, person)* – основне завдання, де перша особа – це свояк, а інша – жінка (тобто ось чоловік і жінка, жінка – це друга особа, у жінки є сестра, а чоловік цієї сестри – це перша особа).

CLAUSES (*clauses*) – розділ, який поділяється на два типи: факти, правила. Про факти вже казали – ними заповнимо базу даних. А вот правила слід прописати. Основні ідеї: *husband(person, person)* – перевірити, що перший чоловік, перевірити, що другий жінка, перевірити, щоб не були однаковими і мали спільну дитину (так, тут основна зв'язка є через дитину); *wife(person, person)* – це той же замій *husband(person, person)*, але перший та другий елемент змінені місцями; *sister(person, person)* – перевірити, що перша людина жінка, також перевірити, чи є спільний один з батьків у першої та другої людини, щоб не були однаковими; *brother_in_law(person, person)* – достатньо перевірити, що є сестра для другої людини (тобто сестра жінки), перша людина має бути чоловіком для цієї сестри, перша і друга особа є різними людьми.

GOAL (*goal*) – розділ, де задається головне питання.

3. Вихідний текст програми розв'язку задачі (основні фрагменти з коментарями)

domains

person=symbol

predicates

nondeterm parent(person, person) % person1 is the parent of the person2.

nondeterm woman(person) % person is the woman.

nondeterm man(person)%person is the man.

nondeterm husband(person,person)%person1 is the husband of the person 2.

nondeterm wife(person,person)%person1 is the wife of the person 2.

nondeterm sister(person,person)%person1 is the sister of the person 2.

nondeterm brother_in_law(person,person)%person1 is the husband of the sister - husband2.

clauses

man(andrii).

man(vova).

man(llya).

man(sasha).

man(temp_son).

woman(lesya).

woman(oksana).

woman(nina).

parent(vova, andrii).

parent(lesya, andrii).

```
parent(vladimir,lesya).  
parent(vladimir,oksana).  
parent(vladimir,llya).
```

```
parent(nina,lesya).  
parent(nina,oksana).  
parent(nina,llya).
```

```
parent(oksana,temp_son).  
parent(sasha,temp_son).
```

```
husband(X,Y):-man(X),woman(Y),parent(X,Z),parent(Y,Z),X<>Y.
```

```
wife(X,Y):-husband(Y,X).
```

```
sister(X,Y):-woman(X),parent(Z,X),parent(Z,Y),X<>Y.
```

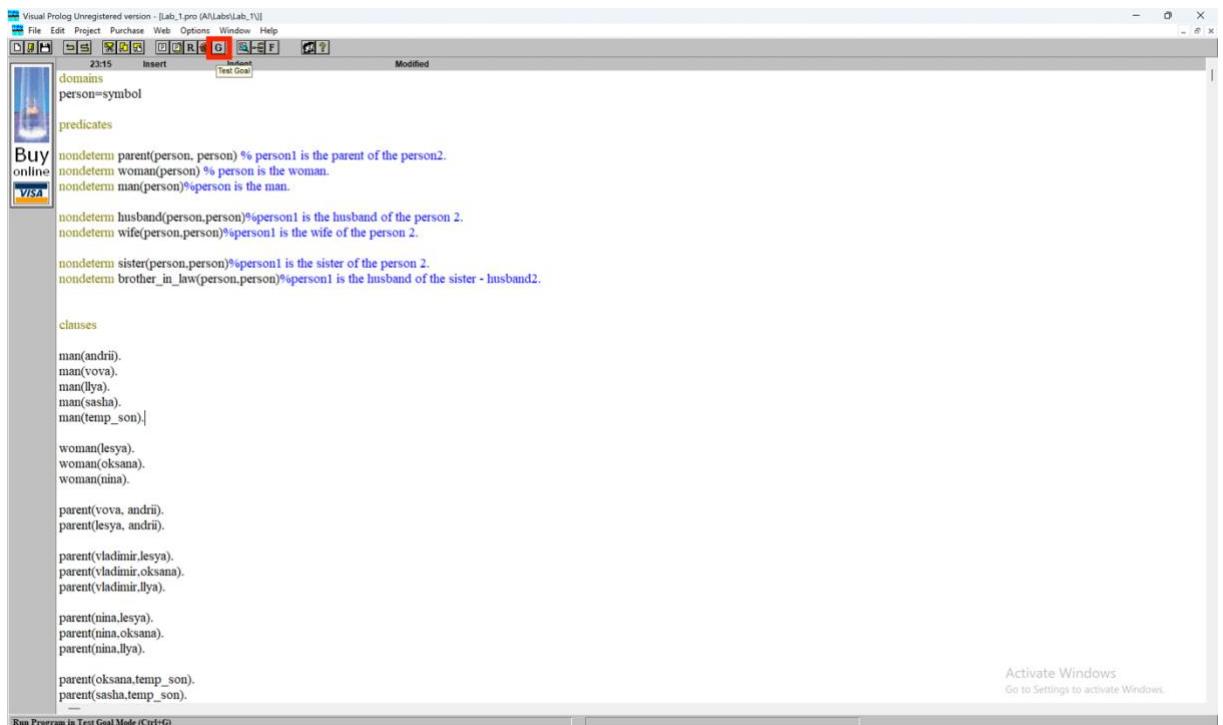
```
brother_in_law(X,Y):-sister(T,Y),husband(X,T),woman(Y),X<>Y.
```

```
goal
```

```
brother_in_law(A,B).
```

4. Опис інтерфейсу програми (керівництво користувача)

Після написання коду – можна запустити тест на виконання за допомогою кнопки, яка показано на рисунку (Рисунок 2).



The screenshot shows the Visual Prolog IDE interface. The menu bar includes File, Edit, Project, Purchase, Web, Options, Windows, Help, and a highlighted Test Goal button. The title bar says "Visual Prolog Unregistered version - [Lab_1.pro (All)]". The code area contains:

```
domains
person=symbol

predicates
nondet parent(person, person) % person1 is the parent of the person2.
nondet woman(person) % person is the woman.
nondet man(person)%person is the man.

nondet husband(person,person)%person1 is the husband of the person 2.
nondet wife(person,person)%person1 is the wife of the person 2.

nondet sister(person,person)%person1 is the sister of the person 2.
nondet brother_in_law(person,person)%person1 is the husband of the sister - husband2.

clauses
man(andrii).
man(vova).
man(ilya).
man(sasha).
man(temp_son)|

woman(lesya).
woman(oksana).
woman(nina).

parent(vova, andrii).
parent(lesya, andrii).

parent(vladimir,lesya).
parent(vladimir,oksana).
parent(vladimir,ilya).

parent(nina,lesya).
parent(nina,oksana).
parent(nina,ilya).

parent(oksana,temp_son).
parent(sasha,temp_son).
—
```

At the bottom right, there is an "Activate Windows" watermark.

Рисунок 2 – Інтерфейс користувача та кнопка запуску теста.

5. Опис тестових прикладів

Запустимо тест програми.



Рисунок 3 – Запуск теста.

Можна побачити, що видано правильні результати, але є повтори. Наразі невідомо, як їх позбудитися, але в подальшому навченні є надія, що повтори можна прибрати.

6. Виправлення помилок

6.1 Хід усунення помилок

Після консультації з керівником – Золотко Костянтин Євгенович, було прийнято рішення покращити код. Було сказано, що проблема, найімовірніше, у правилі.

Роботу розпочали з нуля.

Першим правилом, яке розглядалося, було $sister(X,Y)$ - тобто особа X є сестрою особи Y. Основна перевірка: X - жінка, X та Y мають одного спільногого з батьків – жінка (щоб не було дублікатів), X та Y - різні люди, спільна мати не є ні X, ні Y.

Дві сестри можна назвати сестрами, якщо одна є сестрою другої, а друга - першої. Тобто, введемо правило $sisters(X,Y)$ - де припускаємо, що X є сестрою Y, а водночас Y є сестрою X.

Правило для чоловіка міститиме самого чоловіка та його дружину: $husband(X,Y)$. Спочатку нам потрібно перевірити, що X - чоловік, а Y - жінка, тоді X та Y повинні мати одну спільну дитину, і жодна особа не може бути іншим одночасно, X не є Y, і ні X, ні Y не є дитиною.

І нарешті, свояк: $brother_in_law(X,Y)$ – свояк та дружина. Ми перевіряємо, чи є у Y сестра, і чи є вона сестрою, а потім, для сестри Y, ми перевіряємо, чи є X чоловіком.

6.2 Код програми

domains

person=string

predicates

nondeterm equal(person,person)
nondeterm not_equal(person,person)

nondeterm man(person)

nondeterm woman(person)

nondeterm parent(person, person)

nondeterm sister(person, person)

nondeterm sisters(person, person)

nondeterm husband(person, person)

nondeterm brother_in_law(person, person)

clauses

man("Andrii").

man("Vladimir").

man("Vova").

man("Iilay").

man("Sasha").

woman("Lesya").

woman("Nina").

woman("Oksana").

woman("Kate").

parent("Vladimir", "Andrii").

parent("Lesya", "Andrii").

parent("Nina", "Lesya").

parent("Nina", "Oksana").

parent("Nina", "Iilay").

parent("Vova", "Lesya").

parent("Vova", "Oksana").

parent("Vova", "Iilay").

parent("Oksana", "Kate").

parent("Sasha", "Kate").

```

equal(X,Y):-X=Y.
not_equal(X,Y):-X<>Y.

```

```

sister(X,Y):-
woman(X),parent(Z,X),parent(Z,Y),woman(Z),not_equal(X,Y),not_equal(X,Z),not_equal(Y,Z).

sisters(X,Y):-sister(X,Y),sister(Y,X).

```

```

husband(X,Y):-
man(X),woman(Y),parent(X,Z),parent(Y,Z),not_equal(X,Y),not_equal(X,Z),not_equal(Y,Z).

brother_in_law(X,Y):-sisters(Y,Z),husband(X,Z),not_equal(X,Y),not_equal(X,Z),not_equal(Y,Z).

```

```

goal
brother_in_law(A,B).

```

6.3 Прикла роботи програми

```

Visual Prolog Unregistered version - [Lab_1.pro (All\Lab\Lab_1)]
File Edit Project Purchase Web Options Window Help
Buy online VISA
clauses
man("Andrii").
man("Vladimir").
man("Vova").
man("Ilay").
man("Sasha").

woman("Lesya").
woman("Nina").
woman("Oksana").
woman("Kate").

parent("Vladimir","Andrii").
parent("Lesya","Andrii").

parent("Nina","Lesya").
parent("Nina","Oksana").
parent("Nina","Ilay").

parent("Vova","Lesya").
parent("Vova","Oksana").
parent("Vova","Ilay").

parent("Oksana","Kate").
parent("Sasha","Kate").

equal(X,Y):-X=Y.
not_equal(X,Y):-X<>Y.

sister(X,Y):-woman(X),parent(Z,X),parent(Z,Y),woman(Z),not_equal(X,Y),not_equal(X,Z),not_equal(Y,Z).

sisters(X,Y):-sister(X,Y),sister(Y,X).

husband(X,Y):-man(X),woman(Y),parent(X,Z),parent(Y,Z),not_equal(X,Y),not_equal(X,Z),not_equal(Y,Z).

brother_in_law(X,Y):-sisters(Y,Z),husband(X,Z),not_equal(X,Y),not_equal(X,Z),not_equal(Y,Z).

goal
brother_in_law(A,B).

```

Activate Windows
Go to Settings to activate Windows.

Рисунок 4 – Факти,правила та запит програми.



Рисунок 5 – Результат роботи програми.

Можна побачити, що видано правильні результати, без повторів.