

Национальный исследовательский университет информационных  
технологий, механики и оптики  
Факультет ПИиКТ

Системы искусственного интеллекта  
Лабораторная работа №1

Работу выполнил:  
Конаныхина А.А.

Группа:  
Р33102

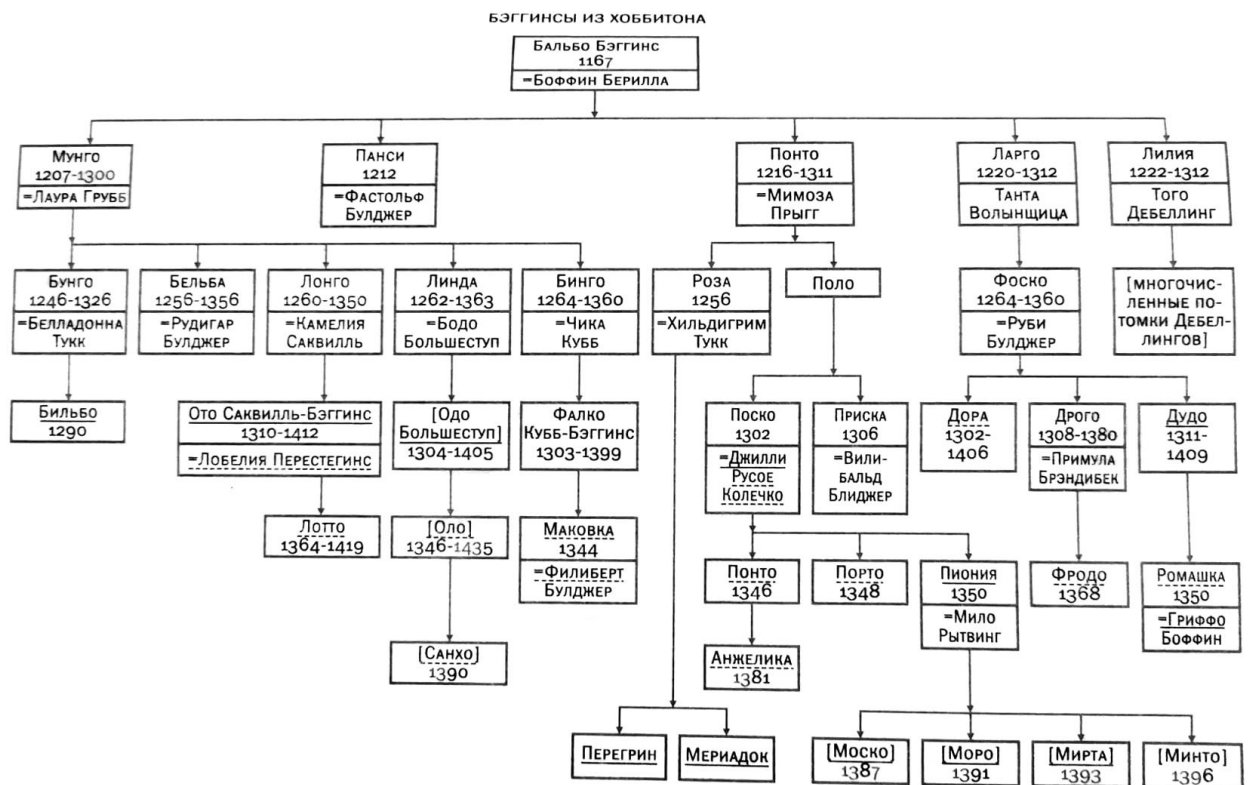
Преподаватель:  
Кугаевских А.В.

Санкт-Петербург  
2022

## Задание:

Составить базу знаний на языке Пролог, которая будет содержать генеалогическое дерево семьи с указанием родственных отношений (например, папа, бабушка и т.д.), опираясь на два первичных факта: родитель и супруг. База знаний должна состоять не менее чем из 50 фактов и не менее чем 10 правил.

## Выполнение лабораторной:



## Код:

% База знаний

marrieds('Бальбо Бэггинс', 'Боффин Берилла').  
marrieds('Мунго Бэггинс', 'Лаура Грубб').  
marrieds('Панси Бэггинс', 'Фастольф Булджер').  
marrieds('Понто Бэггинс', 'Мимоза Прыгг').  
marrieds('Ларго Бэггинс', 'Танта Волынщица').  
marrieds('Лилия Бэггинс', 'Того Дебеллинг').  
marrieds('Бунго Бэггинс', 'Белладонна Тукк').  
marrieds('Бельба Бэггинс', 'Рудигар Булджер').  
marrieds('Лонга Бэггинс', 'Камелия Саквилль').  
marrieds('Линда Бэггинс', 'Бодо Большеступ').  
marrieds('Бинго Бэггинс', 'Чика Кубб').  
marrieds('Роза Бэггинс', 'Хильдрим Тукк').  
marrieds('Фоско Бэггинс', 'Руби Бульджер').  
marrieds('Ото Саквилль-Бэггинс', 'Лобелия Перестегинс').  
marrieds('Поско Бэггинс', 'Джилли Русое Колечко').

marrieds('Приска Бэггинс', 'Виллибальд Блиджер').  
marrieds('Дрого Бэггинс', 'Примула Брэндибек').  
marrieds('Маковка Бэггинс', 'Филиберт Бульджер').  
marrieds('Пиония Бэггинс', 'Мило Рывтинг').  
marrieds('Ромашка Бэггинс', 'Гриффо Боффин').

children('Мунго Бэггинс', 'Бальбо Бэггинс').  
children('Панси Бэггинс', 'Бальбо Бэггинс').  
children('Понто Бэггинс', 'Бальбо Бэггинс').  
children('Ларго Бэггинс', 'Бальбо Бэггинс').  
children('Лилия Бэггинс', 'Бальбо Бэггинс').  
children('Бунго Бэггинс', 'Мунго Бэггинс').  
children('Бельба Бэггинс', 'Мунго Бэггинс').  
children('Лонго Бэггинс', 'Мунго Бэггинс').  
children('Линда Бэггинс', 'Мунго Бэггинс').  
children('Бинго Бэггинс', 'Мунго Бэггинс').  
children('Поло Бэггинс', 'Понто Бэггинс').  
children('Роза Бэггинс', 'Понто Бэггинс').  
children('Фоско Бэггинс', 'Ларго Бэггинс').  
children('Бильбо Бэггинс', 'Бунго Бэггинс').  
children('Ото Саквилль-Бэггинс', 'Лонго Бэггинс').  
children('Одо Большеступ', 'Линда Бэггинс').  
children('Фалко Кубб-Бэггинс', 'Бинго Бэггинс').  
children('Перегрин Тукк', 'Роза Бэггинс').  
children('Мериадок Тукк', 'Роза Бэггинс').  
children('Поско Бэггинс', 'Поло Бэггинс').  
children('Приска Бэггинс', 'Поло Бэггинс').  
children('Дора Бэггинс', 'Фоско Бэггинс').  
children('Дрого Бэггинс', 'Фоско Бэггинс').  
children('Дудо Бэггинс', 'Фоско Бэггинс').  
children('Понто Второй Бэггинс', 'Поско Бэггинс').  
children('Порто Бэггинс', 'Поско Бэггинс').  
children('Пиония Бэггинс', 'Поско Бэггинс').  
children('Фродо Бэггинс', 'Дрого Бэггинс').  
children('Ромашка Бэггинс', 'Дудо Бэггинс').  
children('Анжелика Бэггинс', 'Понто Второй Бэггинс').  
children('Моско Рывтинг', 'Пиония Бэггинс').  
children('Моро Рывтинг', 'Пиония Бэггинс').  
children('Мирта Рывтинг', 'Пиония Бэггинс').  
children('Минто Рывтинг', 'Пиония Бэггинс').  
children('Лото Саквилль-Бэггинс', 'Ото Саквилль-Бэггинс').  
children('Оло Большеступ', 'Одо Большеступ').  
children('Маковка Кубб-Бэггинс', 'Фалко Кубб-Бэггинс').  
children('Санхо Большеступ', 'Оло Большеступ').

% Правила

children(C, P) :-  
 marrieds(P1, P), children(C, P1), !.

grandparent(C, GP) :-  
 children(C, P), children(P, GP).

siblings(C1, C2) :-  
    children(C1, P), children(C2, P), C1 \= C2.

uncles(C, U) :-  
    children(C, P), siblings(P, U).

cousins(C, Co) :-  
    uncles(C, U), children(Co, U).

is\_married(P1, P2) :-  
    marrieds(P1, P2).  
is\_married(P1, P2) :-  
    marrieds(P2, P1).

ancestor(C, A) :-  
    children(C, A).  
ancestor(C, A) :-  
    children(C, C1),  
    ancestor(C1, A).

generation\_diff\_help(C, A, N, N1) :-  
    children(C, A), N is N1.  
generation\_diff\_help(C, A, N, N1) :-  
    N2 is N1 + 1,  
    children(C, C1),  
    generation\_diff\_help(C1, A, N, N2).  
generation\_diff(C, A, N) :-  
    generation\_diff\_help(C, A, N, 1).

in\_one\_generation(C, C1) :-  
    generation\_diff(C, 'Бальбо Бэггинс', N),  
    generation\_diff(C1, 'Бальбо Бэггинс', N),  
    C1 \= C.

grandgreatparents(C, GGP) :-  
    generation\_diff(C, GGP, 3).

## Результаты проверки фактов и правил:

1)  
? - children('Поло Бэггинс', P).  
    P = 'Понто Бэггинс'  
    P = 'Мимоза Прыгг'

? - children(C, 'Поло Бэггинс').  
    C = 'Поско Бэггинс'  
    C = 'Приска Бэггинс'

2)

? - grandparent('Маковка Кубб-Бэггинс', GP).

GP = 'Бинго Бэггинс'

GP = 'Чика Кубб'

3)

? - siblings('Дудо Бэггинс', 'Дрого Бэггинс').

True

? - siblings('Бунго Бэггинс', S2).

S2 = 'Бельба Бэггинс'

S2 = 'Лонго Бэггинс'

S2 = 'Линда Бэггинс'

S2 = 'Бинго Бэггинс'

4)

? - uncles('Приска Бэггинс', U).

U = 'Роза Бэггинс'

? - uncles(C, 'Поло Бэггинс').

C = 'Перегрин Тукк'

C = 'Мериадок Тукк'

5)

? - cousins('Бельба Бэггинс', Co).

Co = 'Поло Бэггинс'

Co = 'Роза Бэггинс'

Co = 'Фоско Бэггинс'

6)

? - is\_married('Роза Бэггинс', 'Хильгдрим Тукк').

true

7)

? - ancestor('Фалко Кубб-Бэггинс', A).

A = 'Бинго Бэггинс'

A = 'Чика Кубб'

A = 'Мунго Бэггинс'

A = 'Лаура Грубб'

A = 'Бальбо Бэггинс'

A = 'Боффин Берилла'

? - ancestor(C, 'Фоско Бэггинс').

C = 'Дора Бэггинс'

C = 'Дрого Бэггинс'

C = 'Дудо Бэггинс'

C = 'Фродо Бэггинс'

C = 'Ромашка Бэггинс'

8)

? - generation\_diff('Дрого Бэггинс', 'Бальбо Бэггинс', N).

N = 3

? - generation\_diff('Дрого Бэггинс', A, 2).

A = 'Ларго Бэггинс'

A = 'Танта Волынщица'

9)

? - in\_one\_generation('Анжелика Бэггинс', C1).

C1 = 'Моско Рывтинг'

C1 = 'Моро Рывтинг'

C1 = 'Мирта Рывтинг'

C1 = 'Минто Рывтинг'

C1 = 'Санхо Большеступ'

10)

? - grandgreatparents('Поско Бэггинс', GGP).

GGP = 'Бальбо Бэггинс'

GGP = 'Боффин Берилла'

### **Вывод:**

При выполнении данной лабораторной работы мною были изучены основы синтаксиса и базовые понятия логического языка программирования Пролог, такие как факты и правила, и написана программа, соответствующая поставленной задаче данной лабораторной.