**Министерство науки и высшего образования Российской Федерации**

**федеральное государственное автономное образовательное учреждение высшего образования**

**“НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО”**

|  |  |
| --- | --- |
| **Факультет** | **Программной Инженерии и Компьютерной Техники** |
| **Направление подготовки (специальность)** |  |
| **Дисциплина** | **Системы искусственного интеллекта** |

Лабораторная работа 1

ОТЧЕТ

|  |  |
| --- | --- |
| **Выполнил студент:** | **Алхимовици Арсений (408138)** |
| **Группа:** | **P3210** |
| **Преподаватель:** | **Болдырева Елена Александровна (157150)** |

г. Санкт-Петербург

2025

Содержание

[***ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ*** 2](#_Toc207782920)

[***ОТЧЕТ О ХОДЕ ВЫПОЛНЕНИЯ*** 2](#_Toc207782921)

[***СПИСОК ФАКТОВ С ОПИСАНИЕМ АРГУМЕНТОВ:*** 3](#_Toc207782922)

[***СПИСОК ПРАВИЛ С КОММЕНТАРИЯМИ:*** 4](#_Toc207782923)

[***ПРИМЕРЫ ВЫПОЛНЕНИЯ ЗАПРОСОВ:*** 7](#_Toc207782924)

[***ЗАКЛЮЧЕНИЕ*** 13](#_Toc207782925)

# ***ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ***

Создание базы знаний в среде Prolog

Предметная область – Ваше семейное дерево

1. Создать не менее 30 объектов – членов вашей семьи.
2. Факты, отражающие состояние – события, регистрируемые органами ЗАГС (рождение, смерть, заключение брака, расторжение брака).
3. Создать не менее 30 правил, определяющих на основании событий состояния членов семьи и отношения между ними.
4. Правила должны учитывать темпоральность состояний с точностью до года.

Содержание отчета:

1. Семейное дерево (можно хоть от руки).
2. Список фактов с описанием аргументов.
3. Список правил с комментариями.
4. Не менее 10 скриншотов с результатами выполнения запросов. Запросы могут быть примерно такими: На ком был ХХХ женат в 1997 году? Запросы НЕ на естественном языке. На Прологе.

Если у вас не набирается 30 человек, можете выдумать недостающих.

Если не хотите раскрывать сведения о своей семье, можете частично анонимизировать данные. Но когда станете знаменитостью, то можете сильно упростить работу своим биографам ☺. Мы же распространять ваши данные не будем.

.

# ***ОТЧЕТ О ХОДЕ ВЫПОЛНЕНИЯ***

Построенное семейное дерево (представлено на рисунке 1):

A diagram of a flowchart

AI-generated content may be incorrect.

*Рисунок 1 – Семейное дерево*

# ***СПИСОК ФАКТОВ С ОПИСАНИЕМ АРГУМЕНТОВ:***

**born/3:** рождение.

* Name: имя человека.
* Year: год рождения.
* Sex: пол человека.

**died/2:** смерть.

* Name: имя человека.
* Year: год смерти.

**married/3:** заключение брака.

* Name1: имя первого человека.
* Name2: имя второго человека.
* Year: год заключения брака.

**divorced/3:** расторжение брака.

* Name1: имя первого человека.
* Name2: имя второго человека.
* Year: год расторжения брака.

**parents/3:** родители ребенка.

* Parent1: имя первого родителя.
* Parent2: имя второго родителя.
* Child: имя ребенка.

# ***СПИСОК ПРАВИЛ С КОММЕНТАРИЯМИ:***

|  |
| --- |
| % жив ли такой то персонаж в таком то году  alive(Person, Year) :-  number(Year),  Year =< 2025,  born(Person, BirthYear, \_),  BirthYear =< Year,  ( \+ died(Person, \_) ; (died(Person, DiedYear), DiedYear > Year)).  % true, если Parent является одним из родителей Child  parent\_of(P, C) :-  parents(P, \_, C) ; parents(\_, P, C).    % человек - мужчина?  male(Person) :-  born(Person, \_, m).  % человек - женщина?  female(Person) :-  born(Person, \_, f).  % нахождение отца ребенка  father(F, Child) :-  parent\_of(F, Child),  male(F).  % нахождение матери ребенка  mother(M, Child) :-  parent\_of(M, Child),  female(M).  % полные братья или сестры: (оба родителя)  full\_sibling(X, Y) :-  X \= Y,  mother(M, X), mother(M, Y),  father(F, X), father(F, Y).  % один общий родитель  half\_sibling(X, Y) :-  X \= Y,  (  (mother(M, X), mother(M, Y), \+ full\_sibling(X, Y))  ;  (father(F, X), father(F, Y), \+ full\_sibling(X, Y))  ).  % хотя бы 1 общий родитель  sibling(X, Y) :-  X \= Y,  parent\_of(P, X),  parent\_of(P, Y).  % нахождение сестер  sisters(X, Y):-  female(X),  female(Y),  full\_sibling(X, Y).  % нахождение братьев  brothers(X, Y):-  male(X),  male(Y),  full\_sibling(X, Y).    % женаты ли всю жизнь  is\_married(Wife, Husband):-  married(Wife, Husband, \_) ; married(Husband, Wife, \_),  \+ divorced(Wife, Husband, \_) ; \+ divorced(Husband, Wife, \_).  % найти мужа  husband(Wife, Husband):-  female(Wife),  is\_married(Wife, Husband),  male(Husband).  % найти жену  wife(Husband, Wife):-  male(Husband),  is\_married(Husband, Wife),  female(Wife).  % двоюродные братья и сестры  cousins(X, Y):-  \+ sibling(X, Y),  parent\_of(P1, X),  parent\_of(P2, Y),  full\_sibling(P1, P2).  % двоюродные сестры  cousins\_sisters(X, Y):-  female(X),  female(Y),  cousins(X, Y).  % двоюродные братья  cousins\_brothers(X, Y):-  male(X),  male(Y),  cousins(X, Y).  % бабушки дедушки  grandparent(Child, Grandparent) :-  parent\_of(Parent, Child),  parent\_of(Grandparent, Parent).  % бабушки  grandmother(Child, Grandmother):-  female(Grandmother),  grandparent(Child, Grandmother).  % дедушки  grandfather(Child, Grandfather):-  male(Grandfather),  grandparent(Child, Grandfather).  % универсальная проверка на брак (без разницы кто идет первый)  married\_univ(X, Y, Year):-  married(X, Y, Year) ; married(Y, X, Year).  % женаты ли в такой год  married\_in(X, Y, Year):-  married\_univ(X, Y, MarryYear),  Year >= MarryYear,  (\+divorced(X, Y, \_) ; (divorced(X, Y, DivYear), DivYear > Year)).  % дети в такой то год  childrens\_in(Parent, Child, Year):-  born(Child, BornYear, \_),  BornYear =< Year,  parent\_of(Parent, Child).  % сыновья в такой то год  sons\_in(Parent, Boy, Year):-  male(Boy),  childrens\_in(Parent, Boy, Year).  % дочери в такой то год  daugthers\_in(Parent, Girl, Year):-  male(Girl),  childrens\_in(Parent, Girl, Year).  % сирота ли в такой то год  orphan\_in(Child, Year):-  born(Child, BornYear, \_),  BornYear =< Year,  parent\_of(P1, Child),  parent\_of(P2, Child),  P1 \= P2,  died(P1, P1Y),  died(P2, P2Y),  P1Y =< Year,  P2Y =< Year.  % дяди  uncle(Uncle, Child) :-  parent\_of(Parent, Child),  brothers(Uncle, Parent).  % тети  aunt(Aunt, Child) :-  parent\_of(Parent, Child),  sisters(Aunt, Parent).  % внуки внучки  grandchild(Grandparent, Grandchild) :-  parent\_of(Parent, Grandchild),  parent\_of(Grandparent, Parent).  % кол-во детей к такому то году  children\_count\_in(Parent, Year, Count) :-  findall(Child, childrens\_in(Parent, Child, Year), Children),  length(Children, Count).  % кол-во братьев к такому то году  brothers\_count\_in(X, Year, Count) :-  findall(Bro, (brothers(X, Bro), born(Bro, BY, \_), BY =< Year), Bros),  length(Bros, Count).  % кол-во систер к такому то году  sisters\_count\_in(X, Year, Count) :-  findall(Sis, (sisters(X, Sis), born(Sis, BY, \_), BY =< Year), Siss),  length(Siss, Count). |

# ***ПРИМЕРЫ ВЫПОЛНЕНИЯ ЗАПРОСОВ:***

A black background with white text

AI-generated content may be incorrect.

*Рисунок 2 – Запрос “Братья или сестры Арсения”*

A black background with white text

AI-generated content may be incorrect.

*Рисунок 3 – Запрос “Братья Арсения”*

A black screen with white text

AI-generated content may be incorrect.

*Рисунок 4 – Запрос “С кем состояла в браке Светлана в 2007? Состояла ли она в браке в 2010?”*

A computer code on a black background

AI-generated content may be incorrect.

*Рисунок 5 – Запрос “Дети Дмитрия в 2009 и 2005?”*

A black background with white text

AI-generated content may be incorrect.

*Рисунок 6 – Запрос “orphan сирота в 2015?”*

A black background with white text

AI-generated content may be incorrect.

*Рисунок 7 – Запрос “Двоюродные братья/сестры Матвея”*

A computer screen with white text

AI-generated content may be incorrect.

*Рисунок 8 – Запрос “Внуки Александры.”*

A black screen with white text

AI-generated content may be incorrect.

*Рисунок 9 – Запрос “Количество детей у светланы в 2008”*

A black screen with white text

AI-generated content may be incorrect.

*Рисунок 10 – Запрос “Тетя и дядя Арсения?”*

A black background with white text

AI-generated content may be incorrect.

*Рисунок 11 – Запрос “Жив ли Трамп в 2025?”*

# ***2 Часть: Protege***

A screenshot of a computer

AI-generated content may be incorrect.

*Рисунок 12 – Граф Protege.*

A screenshot of a computer

AI-generated content may be incorrect.

*Рисунок 13 – Полученные данные для ulyana\_a после “start Reasoner”*

A screenshot of a computer

AI-generated content may be incorrect.

*Рисунок 14 – Полученные данные для sergey\_s после “start Reasoner”*

A screenshot of a computer

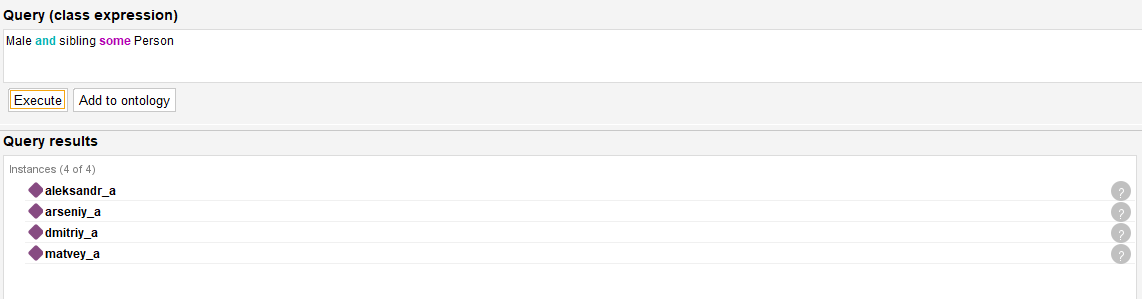
AI-generated content may be incorrect.

*Рисунок 15 – Запрос “Person”*

A screenshot of a computer

AI-generated content may be incorrect.

*Рисунок 16 – Запрос “дети у которых есть дети?”*



*Рисунок 17 – Запрос “Мужчины у которых есть братья/сестры”*

A screenshot of a computer

AI-generated content may be incorrect.

*Рисунок 18 – Запрос “Человек, который брат или сестра кого-то при этом мать кого-то.”*

A screenshot of a computer

AI-generated content may be incorrect.

*Рисунок 19 – SWRL правила*

# ***ЗАКЛЮЧЕНИЕ***

В ходе лабораторной работы была реализована база знаний на Prolog, моделирующая семейное дерево с учётом жизненных событий (рождения, браки, разводы, смерти). На основе фактов и правил стало возможным формализованно определять родственные отношения и отслеживать изменения статуса членов семьи во времени.

Практическим результатом стало то, что система не только хранит структурированные данные о семье, но и делает выводы, недоступные при простом перечислении. Работа продемонстрировала ценность логического программирования для представления знаний и рассуждений в задачах, где важна причинно-временная последовательность.

***2 Часть:***

В ходе лабораторной работы в средеProtege была построена онтология семейного дерева с использованием классов, свойств и правил SWRL. Reasoner позволил автоматически выводить новые родственные отношения (брат, сестра, мать, дядя и др.), а не только хранить исходные факты. Практическим результатом стало получение системы, которая проверяет согласованность данных и расширяет знания о семье за счёт логического вывода.