Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования Московский государственный технический университет имени Н.Э. Баумана

(национальный исследовательский университет) (МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

НАПРАВЛЕНИЕ ПОДГОТОВКИ «09.03.04 Программная инженерия»

ОТЧЕТ

по лабораторной работе №9

Название: Использование правил в программе на Prolog

Дисциплина: Функциональное и логическое программирование

|  |  |  |  |
| --- | --- | --- | --- |
| Студент | ИУ7-65Б  Группа | Подпись, дата | Талышева О.Н.  Фамилия И.О. |
| Преподаватель |  | Подпись, дата | Толпинская Н.Б.  Фамилия И.О. |

Москва, 2025 г.

1. **Задание: База знаний «Предки»**

Создать базу знаний «Предки», позволяющую наиболее эффективным способом (за меньшее количество шагов, что обеспечивается меньшим количеством предложений БЗ - правил), и используя разные варианты (примеры) простого вопроса, (указать: какой вопрос для какого варианта) определить:

1. по имени субъекта определить всех его бабушек (предки 2-го колена),
2. по имени субъекта определить всех ого дедушек (предки 2-го колена),
3. по имени субъекта определить всех его бабушек и дедушек (предки 2-го
4. колена),
5. по имени субъекта определить его бабушку по материнской линии (предки 2-го колена),
6. по имени субъекта определить его бабушку и дедушку по материнской линии. (предки 2-го колена).

Минимизировать количество правил и количество вариантов вопросов. Использовать конъюнктивные правила и простой вопрос.

**Результаты работы:**

Ниже идут листинги программы с несколькими вариантами вопросов с ответами:

**domains**

name = symbol.

gender\_type = male; female.

**predicates**

gender(name, gender\_type).

parent(name, name).

grandparents(name, name, gender\_type, gender\_type).

**clauses**

gender(anna, female). gender(inna, female).

gender(elena, female). gender(daria, female).

gender(olga, female). gender(lisa, female). gender(mia, female).

gender(nik, male). gender(met, male). gender(stefan, male).

gender(timur, male). gender(ivan, male).

gender(tom, male). gender(leo, male).

parent(anna, inna). parent(stefan, inna). parent(inna, daria).

parent(inna, tom). parent(olga, ivan). parent(leo, ivan).

parent(ivan, daria). parent(ivan, tom). parent(met, elena).

parent(elena, nik). parent(lisa, timur). parent(timur, nik). parent(nik, mia).

grandparents(Grandparent, Name, Grandparent\_gender, Parent\_gender) :- parent(Grandparent, Parent), parent(Parent, Name), gender(Parent, Parent\_gender), gender(Grandparent, Grandparent\_gender).

**goal**

|  |  |
| --- | --- |
| grandparents(Name, daria, female, \_). | Name=anna  Name=olga  2 Solutions |
| grandparents(Name, daria, male, \_). | Name=stefan  Name=leo  2 Solutions |
| grandparents(Name, daria, \_, \_). | Name=anna  Name=stefan  Name=olga  Name=leo  2 Solutions |
| grandparents(Name, daria, female, female). | Name=anna  1 Solution |
| grandparents(Name, daria, \_, female). | Name=anna  Name=stefan  2 Solutions |

**2. Дополнить базу знаний правилами, позволяющими найти.**

1. Максимум из двух чисел

а) без использования отсечения,

в) с использованием отсечения;

1. Максимум из трех чисел.

а) без использования отсечения,

в) с использованием отсечения;

**Результаты работы:**

Ниже идут листинги программы с несколькими вариантами вопросов с ответами:

**predicates**

max\_2(**integer**, **integer**, **integer**).

max\_2\_cut(**integer**, **integer**, **integer**).

max\_3(**integer**, **integer**, **integer**, **integer**).

max\_3\_cut(**integer**, **integer**, **integer**, **integer**).

**clauses**

max\_2(X, Y, X) :- X >= Y.

max\_2(X, Y, Y) :- X < Y.

max\_2\_cut(X, Y, X) :- X >= Y, !.

max\_2\_cut(\_, Y, Y).

max\_3(X, Y, Z, X) :- X >= Y, X >= Z.

max\_3(X, Y, Z, Y) :- Y >= X, Y >= Z.

max\_3(X, Y, Z, Z) :- Z >= X, Z >= Y.

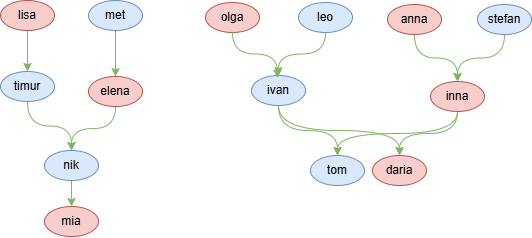
max\_3\_cut(X, Y, Z, X) :- X >= Y, X >= Z, !.

max\_3\_cut(\_, Y, Z, Y) :- Y >= Z, !.

max\_3\_cut(\_, \_, Z, Z).

**goal**

|  |  |
| --- | --- |
| max\_2(1, 2, R). | R=2  1 Solution |
| max\_2\_cut(1, 2, R). | R=2  1 Solution |
| max\_3(1, 3, 2, R). | R=3  1 Solution |
| max\_3\_cut(1, 3, 2, R). | R=3  1 Solution |



**Таблица для цели grandparents(Name, daria, female, \_).**

|  |  |  |  |
| --- | --- | --- | --- |
| **№ шага** | **Состояние резольвенты, и вывод: дальнейшие действия (почему?)** | **Для каких термов запускается алгоритм унификации: T1=T2 и каков результат (и подстановка)** | **Дальнейшие действия: прямой ход или откат (почему и к чему приводит?)** |
| 1 | grandparents(Name, daria, female, \_) | T1 = grandparents(Name, daria, female, \_)  T2 = grandparents(Grandparent, Name, Grandparent\_gender, Parent\_gender)  Результат: Grandparent = Name, Name = daria, Grandparent\_gender = female, Parent\_gender = \_ | Прямой ход: замена цели на подцели из правила. |
| 2 | parent(Grandparent, Parent),  parent(Parent, daria),  gender(Parent, Parent\_gender),  gender(Grandparent, female). | Т1 = parent(Grandparent, Parent)  Т2 = parent(anna, inna).  Подстановка: Grandparent = anna, Parent = inna. | Прямой ход: поиск подходящих фактов. |
| 3 | parent(inna, daria),  gender(inna, Parent\_gender),  gender(anna, female). | Т1 = parent(inna, daria)  Т2 = parent(inna, daria). | Прямой ход: переход к следующей подцели. |
| 4 | gender(inna, Parent\_gender),  gender(anna, female). | Т1 = gender(inna, Parent\_gender)  Т2 = gender(inna, female).  Подстановка: Parent\_gender = female. | Прямой ход: Parent\_gender = female, переход к следующей подцели. |
| 5 | gender(anna, female). | Т1 = gender(anna, female)  Т2 = gender(anna, female). | Успех: все подцели выполнены. Результат: Name = anna. Откат: поиск альтернативных решений. |
| 6 | (Откат для поиска других решений)  parent(Grandparent, Parent),  parent(Parent, daria),  gender(Parent, Parent\_gender),  gender(Grandparent, female). | Возврат к шагу 2 для поиска других фактов  Т1 = parent(Grandparent, Parent)  Т2 = parent(olga, ivan).  Подстановка: Grandparent = olga, Parent = ivan. | Прямой ход: поиск подходящих фактов. |
| 7 | parent(ivan, daria),  gender(ivan, Parent\_gender),  gender(olga, female). | Т1 = parent(ivan, daria)  Т2 = parent(ivan, daria). | Прямой ход: переход к следующей подцели. |
| 8 | gender(ivan, Parent\_gender),  gender(olga, female). | Т1 = gender(ivan, Parent\_gender)  Т2 = gender(ivan, male).  Подстановка: Parent\_gender = male. | Прямой ход: Parent\_gender = male, переход к следующей подцели. |
| 9 | gender(olga, female). | Т1 = gender(olga, female)  Т2 = gender(olga, female). | Успех: все подцели выполнены. Результат: Name = olga.  Откат: поиск альтернативных решений. |
| 10 | (Откат для поиска других решений) | Возврат к шагу 6. Больше фактов parent(Grandparent, Parent) нет. | Завершение: все возможные решения найдены (Name = anna, Name = olga). |

**Вывод:** Name = anna, Name = olga.

**Таблица для цели max\_2(1, 2, R)**

| **№ шага** | **Состояние резольвенты** | **Унификация: T1=T2 (результат и подстановка)** | **Действия (прямой ход / откат)** |
| --- | --- | --- | --- |
| 1 | max\_2(1, 2, R) | Т1 =  max\_2(1, 2, R)  Т2 = max\_2(X, Y, X)  Подстановка: X=1, Y=2, R=X=1 | Проверка условия 1 >= 2 → ложь. Откат. |
| 2 | max\_2(1, 2, R) | Т1 =  max\_2(1, 2, R)  Т2 = max\_2(X, Y, Y)  Подстановка: X=1, Y=2, R=Y=2 | Проверка условия 1 < 2 → истина. Успех. |

**Вывод:** R = 2.

**Таблица для цели max\_2\_cut(1, 2, R)**

| **№ шага** | **Состояние резольвенты** | **Унификация: T1=T2 (результат и подстановка)** | **Действия (прямой ход / откат)** |
| --- | --- | --- | --- |
| 1 | max\_2\_cut(1, 2, R) | Т1 = max\_2\_cut(1, 2, R)  Т2 = max\_2\_cut(X, Y, X)  Подстановка: X=1, Y=2, R=X=1 | Проверка 1 >= 2 → ложь. Откат. |
| 2 | max\_2\_cut(1, 2, R) | Т1 = max\_2\_cut(1, 2, R)  Т2 = max\_2\_cut(\_, Y, Y)  Подстановка: Y=2, R=Y=2 | Успех. |

**Вывод:** R = 2.

**Таблица для цели max\_3(1, 3, 2, R)**

| **№ шага** | **Состояние резольвенты** | **Унификация: T1=T2 (результат и подстановка)** | **Действия (прямой ход / откат)** |
| --- | --- | --- | --- |
| 1 | max\_3(1, 3, 2, R) | Т1 = max\_3(1, 3, 2, R)  Т2 = max\_3(X, Y, Z, X)  Подстановка: X=1, Y=3, Z=2, R=X=1 | Проверка 1 >= 3 и 1 >= 2 → ложь. Откат. |
| 2 | max\_3(1, 3, 2, R) | Т1 = max\_3(1, 3, 2, R)  Т2 = max\_3(X, Y, Z, Y)  Подстановка: X=1, Y=3, Z=2, R=Y=3 | Проверка 3 >= 1 и 3 >= 2 → истина. Успех. |

**Вывод:** R = 3.

**Таблица для цели max\_3\_cut(1, 3, 2, R)**

| **№ шага** | **Состояние резольвенты** | **Унификация: T1=T2 (результат и подстановка)** | **Действия (прямой ход / откат)** |
| --- | --- | --- | --- |
| 1 | max\_3\_cut(1, 3, 2, R) | Т1 = max\_3\_cut(1, 3, 2, R)  Т2 = max\_3\_cut(X, Y, Z, X)  Подстановка: X=1, Y=3, Z=2, R=X=1 | Проверка 1 >= 3 и 1 >= 2 → ложь. Откат. |
| 2 | max\_3\_cut(1, 3, 2, R) | Т1 = max\_3\_cut(1, 3, 2, R)  Т2 = max\_3\_cut(\_, Y, Z, Y)  Подстановка: Y=3, Z=2, R=Y=3 | Проверка 3 >= 2 → истина. Отсечение ! предотвращает откат. Успех. |

**Вывод:** R = 3.