|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное 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» (ИУ7)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.03.04 Программная инженерия**

**ОТЧЕТ**

|  |  |
| --- | --- |
| **по лабораторной работе №** | 14 |
|  |  |

**Дисциплина:** Функциональное и логическое программирование

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ7-62Б |  |  | И. В. Козлова |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
| Преподаватель |  |  |  | Н.Б.Толпинская |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |
| Преподаватель |  |  |  | Ю.В.Строганов |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2022

**Лабораторная работа 14**

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

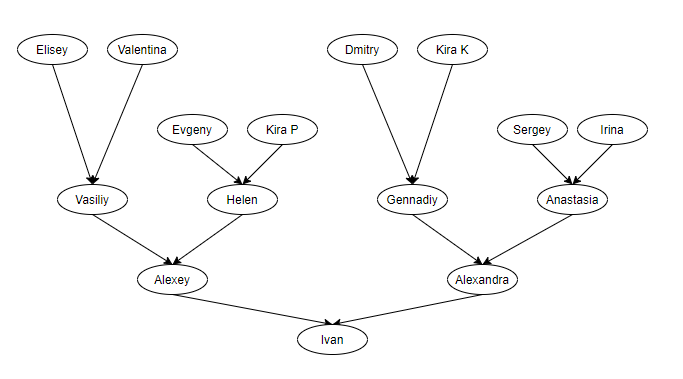
1. По имени субъекта определить всех его бабушек (предки 2-го колена);
2. По имени субъекта определить всех его дедушек (предки 2-го колена);
3. По имени субъекта определить всех его бабушек и дедушек (предки 2-го колена);
4. По имени субъекта определить его бабушку по материнской линии (предки 2-го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

Для одного из вариантов ВОПРОСА и конкретной БЗ составить таблицу, отражающую конкретный порядок работы системы, с объяснениями:

* очередная проблема на каждом шаге и метод ее решения,
* каково новое текущее состояние резольвенты, как получено,
* какие дальнейшие действия? (запускается ли алгоритм унификации? Каких термов? Почему этих?),
* вывод по результатам очередного шага и дальнейшие действия.

Так как резольвента хранится в виде стека, то состояние резольвенты требуется отображать в столбик: вершина – сверху! Новый шаг надо начинать с нового состояния резольвенты!



**Код программы:**

|  |
| --- |
| domains  parent = parent\_st(symbol, symbol).  predicates  isParent(parent, symbol).  ancestors(symbol, symbol, symbol, symbol, symbol).  clauses  isParent(parent\_st(alexey, m), ivan).  isParent(parent\_st(alexandra, w), ivan).    isParent(parent\_st(helen, w), alexey).  isParent(parent\_st(vasiliy, m), alexey).  isParent(parent\_st(kiraP, w), helen).  isParent(parent\_st(evheny, m), helen).  isParent(parent\_st(valentina, w), vasiliy).  isParent(parent\_st(elisey, m), vasiliy).  isParent(parent\_st(anastasia, w), alexandra).  isParent(parent\_st(gennadiy, m), alexandra).  isParent(parent\_st(irina, w), anastasia).  isParent(parent\_st(sergey, m), anastasia).  isParent(parent\_st(kiraK, w), gennadiy).  isParent(parent\_st(dmitry, m), gennadiy).  ancestors(ChildName, MsM, MsF, FsM, FsF) :-  isParent(parent\_st(MothersName, w), ChildName), isParent(parent\_st(FathersName, m), ChildName),  isParent(parent\_st(MsM, w), MothersName), isParent(parent\_st(MsF, m), MothersName),  isParent(parent\_st(FsM, w), FathersName), isParent(parent\_st(FsF, m), FathersName).  goal  % 1  % ancestors(ivan, MsMName, \_, FsMName, \_)  % MsMName=anastasia,  % FsMName=helen    % 2  % ancestors(ivan, \_, MsFName, \_, FsFName)  % MsFName=gennadiy,  % FsFName=vasiliy    % 3  % ancestors("Alexey Ivanov", MsMName, MsFName, FsMName, FsFName)  % MsMName=kiraP,  % MsFName=evheny,  % FsMName=valentina,  % FsFName=elisey    % 4  ancestors(alexey, MsMName, \_, \_, \_)  % MsMName=kiraP    % 5  % ancestors("Alexandra Ivanova", MsMName, MsFName, \_, \_)  % MsMName=irina,  % MsFName=sergey  . |
| Таблица для вопроса ancestors("Alexey Ivanov", MsMName, \_, \_, \_)   |  |  |  |  | | --- | --- | --- | --- | | № шага | Состояние резольвенты, и вывод: дальнейшие действия (почему?) | Для каких термов запускается алгоритм унификации: Т1=Т2 и каков результат (и подстановка) | Дальнейшие действия: прямой ход или откат (к чему приводит?) | | 1 | ancestors("Alexey Ivanov", MsMName, \_, \_, \_)  Резольвента непуста, запуск алгоритма унификации для цели и заголовка первого правила из БЗ | ancestors("Alexey Ivanov", MsMName, \_, \_, \_) = isParent(parent\_st("Alexey Ivanov", m), "Ivan Ivanov")  Неуспешна, разные главные функторы | Откат, переход к следующему предложению | |  | -- | -- | -- | |  | ancestors("Alexey Ivanov", MsMName, \_, \_, \_)  Резольвента непуста, запуск алгоритма унификации для цели и заголовка очередного правила из БЗ | ancestors("Alexey Ivanov", MsMName, \_, \_, \_) = ancestors(ChildName, MsM, MsF, FsM, FsF)  Подстновка  {ChildName = "Alexey Ivanov",  MsMName = MsM,  \_ = MsF,  \_ = FsM,  \_ = FsF } | Формируется новое состояние резольвенты.  isParent(parent\_st(MothersName,w), "Alexey Ivanov"), isParent(parent\_st(FathersName,m), "Alexey Ivanov"),  isParent(parent\_st(MsM, w), MothersName), isParent(parent\_st(MsF, m), MothersName),  isParent(parent\_st(FsM, w), FathersName), isParent(parent\_st(FsF, m), FathersName) | |  | -- | -- | -- | |  | isParent(parent\_st(MothersName,w), "Alexey Ivanov"), isParent(parent\_st(FathersName,m), "Alexey Ivanov"),  isParent(parent\_st(MsM, w), MothersName), isParent(parent\_st(MsF, m), MothersName),  isParent(parent\_st(FsM, w), FathersName), isParent(parent\_st(FsF, m), FathersName)  Резольвента непуста, запуск алгоритма унификации для цели и заголовка очередного правила из БЗ | isParent(parent\_st  (MothersName,w), "Alexey Ivanov") = isParent(parent\_st("Helen Ivanova", w), "Alexey Ivanov")  Подстновка  {ChildName = "Alexey Ivanov",  MsMName = MsM,  \_ = MsF,  \_ = FsM,  \_ = FsF } | Формируется новое состояние резольвенты  isParent(parent\_st(FathersName,m), "Alexey Ivanov"),  isParent(parent\_st(MsM, w), "Helen Ivanova"),  isParent(parent\_st(MsF, m), "Helen Ivanova"),  isParent(parent\_st(FsM, w), FathersName), isParent(parent\_st(FsF, m), FathersName) | |  | -- | -- | -- | |  | isParent(parent\_st(FathersName,m), "Alexey Ivanov"),  isParent(parent\_st(MsM, w), "Helen Ivanova"),  isParent(parent\_st(MsF, m), "Helen Ivanova"),  isParent(parent\_st(FsM, w), FathersName), isParent(parent\_st(FsF, m), FathersName)  Резольвента непуста, запуск алгоритма унификации для цели и заголовка очередного правила из БЗ | isParent(parent\_st(FathersName,m), "Alexey Ivanov") = isParent(parent\_st("Vasiliy Ivanov", m), "Alexey Ivanov  Подстновка  {ChildName = "Alexey Ivanov",  MsMName = MsM,  \_ = MsF,  \_ = FsM,  \_ = FsF } | Формируется новое состояние резольвенты  isParent(parent\_st(MsM, w), "Helen Ivanova"),  isParent(parent\_st(MsF, m), "Helen Ivanova"),  isParent(parent\_st(FsM, w), "Vasiliy Ivanov"), isParent(parent\_st(FsF, m), "Vasiliy Ivanov") | |  | -- | -- | -- | |  | isParent(parent\_st(MsM, w), "Helen Ivanova"),  isParent(parent\_st(MsF, m), "Helen Ivanova"),  isParent(parent\_st(FsM, w), "Vasiliy Ivanov"), isParent(parent\_st(FsF, m), "Vasiliy Ivanov")  Резольвента непуста, запуск алгоритма унификации для цели и заголовка очередного правила из БЗ | isParent(parent\_st(MsM, w), "Helen Ivanova") = isParent(parent\_st("Kira Petrak", w), "Helen Ivanova")  Подстновка  {ChildName = "Alexey Ivanov",  MsMName = "Kira Petrak",  \_ = MsF,  \_ = FsM,  \_ = FsF } | Формируется новое состояние резольвенты  isParent(parent\_st(MsF, m), "Helen Ivanova"),  isParent(parent\_st(FsM, w), "Vasiliy Ivanov"), isParent(parent\_st(FsF, m), "Vasiliy Ivanov") | |  | ancestors("Alexey Ivanov", MsMName, \_, \_, \_)  Резольвента непуста, но достигнут конец БЗ, завершение работы системы | -- | Резольвента непуста, но достигнут конец БЗ, завершение работы системы | |