



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»

Лабораторная работа № 16

Дисциплина Функциональное и логическое программирование

Тема Работа программы на Prolog

Студент Куприй А. А.

Группа ИУ7-63Б

Оценка (баллы) _____

Преподаватель Толпинская Н. Б., Строганов Ю. В.

Москва, 2020 г.

Цель работы – получить навыки построения модели предметной области, разработки и оформления программы на Prolog, изучить принципы, логику формирования программы и отдельные шаги выполнения программы на Prolog.

Задачи работы: приобрести навыки декларативного описания предметной области с использованием фактов и правил.

Изучить способы использования термов, переменных, фактов и правил в программе на Prolog, принципы и правила сопоставления и отождествления, порядок унификации.

Задание:

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

по имени субъекта определить всех его бабушек (предки 2-го колена),

по имени субъекта определить всех его дедушек (предки 2-го колена),

по имени субъекта определить всех его бабушек и дедушек (предки 2-го колена),

по имени субъекта определить его бабушку по материнской линии (предки 2-го колена),

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.

Вопросы:

В каком случае система запускает алгоритм унификации?

Система запускает алгоритм унификации автоматически при необходимости что-то доказать.

Каковы назначение и Результат использования алгоритма унификации?

Унификация – механизм логического вывода. Результат – Подстановка.

Какое первое состояние резольвенты?

Заданный вопрос (goal).

Как меняется резольвента?

Преобразования резольвенты выполняются с помощью редукции.

Редукцией цели G с помощью программы P называется замена цели G телом того правила из P, заголовок которого унифицируется с целью.

Новая резольвента образуется в два этапа:

1. в текущей резольвенте выбирается одна из подцелей и для неё выполняется редукция
2. к полученной конъюнкции целей применяется Подстановка, полученная как наибольший общий унификатор цели и заголовка сопоставленного с ней правила.

В каких пределах программы уникальны переменные?

Именованная переменная уникальна в рамках предложения, в котором она используется.

Как применяется Подстановка, полученная с помощью алгоритма унификации?

Полученная с помощью алгоритма унификации Подстановка применяется к целям в резольвенте.

В каких случаях запускается механизм отката?

Механизм отката запустится в случае неудачи алгоритма унификации.

Текст программы

domains

name = symbol.

flag = symbol.

predicates

/* mother or father, mother, father, child*/

grandparents(flag, name, name, name).

/*mother, father, child*/

parents(name, name, name).

clauses

/* 4th gen*/

parents(nastya, tolya, misha).

parents(nastya, tolya, polina).

parents(varvara, tolya, vitya).

/* 3rd gen*/

parents(ulya, igor, katya).

parents(ulya, igor, zhora).

parents(varvara, andrey, nadya).

/* 2nd gen*/

parents(nadya, misha, dima).

parents(katya, vitya, sasha).

parents(nadya, zhora, rolan).

/* 1st gen*/

parents(sasha, rolan, nikita).

parents(sasha, dima, olya).

grandparents(mom, GrandMa, GrandPa, Child)

:- parents(Mother, _, Child),

parents(GrandMa, GrandPa, Mother).

grandparents(dad, GrandMa, GrandPa, Child) :-

parents(_, Father, Child),

parents(GrandMa, GrandPa, Father).

goal

%grandparents(_, GrandMa, GrandPa, rolan).

%grandparents(_, _, GrandPa, olya).

% grandparents(_, GrandMa, _, olya).

% grandparents(mom, GrandMa, _, olya).

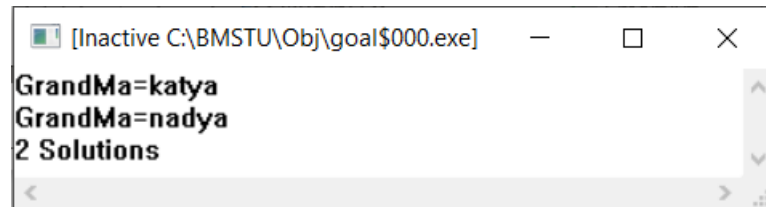
grandparents(mom, GrandMa, _, olya).

Результаты работы программы:

Задание: по имени субъекта определить всех его бабушек (предки 2-го колена)

goal

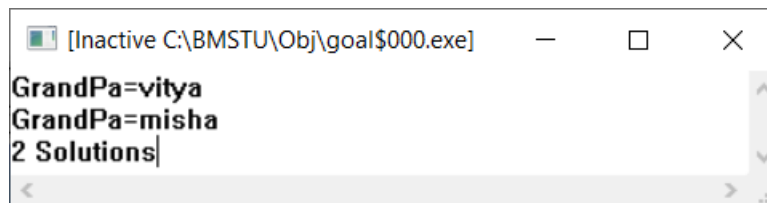
grandparents(, GrandMa, _, olya).



Задание: по имени субъекта определить всех его дедушек (предки 2-го колена)

goal

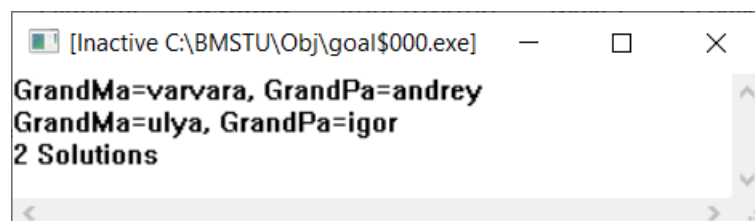
grandparents(, _, GrandPa, olya).



Задание: по имени субъекта определить всех его бабушек и дедушек (предки 2-го колена)

goal

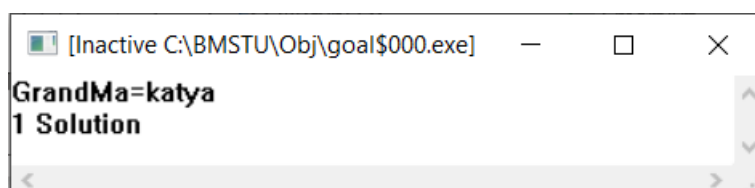
grandparents(, GrandMa, GrandPa, rolan).



Задание: по имени субъекта определить его бабушку по материнской линии (предки 2-го колена)

goal

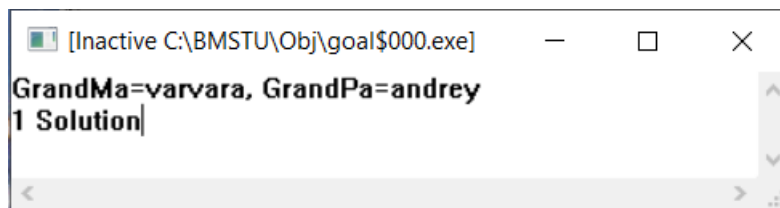
grandparents(mom, GrandMa, _, olya)



Задание: по имени субъекта определить его бабушку и дедушку по материнской линии (предки 2-го колена)

goal

grandparents(mom, GrandMa, GrandPa, rolan).



Описание порядка поиска объектов

Цель: grandparents(mom, GrandMa, _, olya).

| № 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 | Цель заносится в резольвенту. grandparents(mom, GrandMa, _, olya). | Попытка унификации: Grandparents (mom, GrandMa, _, olya) = parents(nastya, tolya, misha). Результат: неудача, разные главные функторы. | Прямой ход, переход к следующему предложению. |
| ... | grandparents(mom, GrandMa, _, olya). | Попытка унификации: Grandparents (mom, GrandMa, _, olya) И термов БЗ, пока не будет найден подходящий терм или не будет «просмотрена» вся БЗ. | Прямой ход (резольвента не пуста, не конец БЗ), переход к следующему предложению. |
| 12 | grandparents(mom, GrandMa, _, olya). | Попытка унификации: Grandparents (mom, GrandMa, _, olya) = grandparents(mom, GrandMa, GrandPa, Name) Результат: успех Подстановка: { GrandMa = GrandMa , Name =olya } | (возможно, установка точки возврата), Преобразование резольвенты (замена текущей цели на тело найденного с помощью унификации правила, применение полученной подстановки), Прямой ход |
| 13 | parents(Mother, _, olya), parents(GrandMa, GrandPa, Mother). Выбор верхней подцели резольвенты. | Попытка унификации: parents(Mother, _, olya) = parents(nastya, tolya, misha). Результат: неудача, несовпадающие константы. | Прямой ход, переход к следующему предложению. |

| | | | |
|-----|---|---|---|
| ... | parents(Mother, _, olya), parents(GrandMa, GrandPa, Mother). | Попытка унификации: parents(Mother, _, olya) и термов, пока не будет найдено подходящее правило или не закончится БЗ. | Прямой ход, переход к следующему предложению. |
| 24 | parents(Mother, _, olya), parents(GrandMa, GrandPa, Mother). | Попытка унификации: parents(Mother, _, olya) = parents(sasha, dima, olya) Результат: успех. Подстановка: {Mother=sasha} | (Возможно, установка точки возврата), Преобразование резольвенты (замена текущей цели на пустое тело найденного с помощью унификации факта, применение полученной подстановки), Прямой ход |
| ... | parents(GrandMa, GrandPa, sasha). Выбор верхней подцели резольвенты. | Попытка унификации: parents(GrandMa, GrandPa, sasha) и термов, пока не будет найдено подходящее правило или не закончится БЗ. | Прямой ход, переход к следующему предложению. |
| 32 | parents(GrandMa, GrandPa, sasha). | Попытка унификации: parents(GrandMa, GrandPa, sasha) = parents(katya, vitya, sasha). Результат: успех Подстановка: {GrandMa=katya, GrandPa=vitya}. | Преобразование резольвенты (замена текущей цели на пустое тело найденного с помощью унификации факта, применение полученной подстановки) |
| 33 | Резольвента пуста. | | Вывод запуск отката |
| 34 | parents(GrandMa, GrandPa, sasha). Выбор верхней подцели резольвенты | Попытка унификации: parents(GrandMa, GrandPa, sasha) = parents(nadya, zhora, rolan) Результат: неудача, несовпадающие константы | Прямой ход, переход к следующему предложению. |

| | | | |
|-----|--|---|--|
| ... | parents(GrandMa, GrandPa, sasha). | Попытка унификации: parents(GrandMa, GrandPa, sasha) и термов, пока не будет найдено подходящее правило или не закончится БЗ. | Прямой ход, переход к следующему предложению |
| 38 | parents(GrandMa, GrandPa, sasha). | Попытка унификации: parents(GrandMa, GrandPa, sasha) = grandparents(dad, GrandMa, GrandPa, Name) Результат: неудача, разные главные функторы. | Конец БЗ, резольвента не пустая, откат. |
| 39 | parents(Mother, _, olya), parents(GrandMa, GrandPa, Mother). Выбор верхней подцели резольвенты. | Попытка унификации: parents(Mother, _, olya) = grandparents(mom, GrandMa, GrandPa, Name) Результат: неудача, разные главные функторы. | Прямой ход, переход к следующему предложению |
| 40 | parents(Mother, _, olya), parents(GrandMa, GrandPa, Mother). | Попытка унификации: parents(Mother, _, olya) = grandparents(dad, GrandMa, GrandPa, Name) Результат: неудача, разные главные функторы. | Конец БЗ, резольвента не пустая, откат. |
| 41 | grandparents(mom, GrandMa, _, olya). | Попытка унификации: Grandparents (mom, GrandMa, _, olya) = grandparents(dad, GrandMa, GrandPa, Name) Результат: неудача, разные константы. | Конец БЗ. Конец работы программы. |