|  |  |
| --- | --- |
|  | **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 № 17**

|  |  |
| --- | --- |
| **Студент:** Керимов А. Ш.  **Группа:** ИУ7-64Б  **Преподаватель:** Толпинская Н. Б. |  |

Москва.

2020 г.

**Цель работы** — изучить способы организации эффективных программ на Prolog, особенности использования системных предикатов и порядок выполнения программ с их использованием.

**Задание.**

**Ответить на вопросы:**

* **Какое первое состояние резольвенты?**

Вопрос

* **В каком случае система запускает алгоритм унификации? (т.е. Как эту необходимость на формальном уровне распознает система?)**

Пролог выполняет унификацию в двух случаях: когда цель сопоставляется с заголовком предложения или когда используется знак равенства, который является инфиксным предикатом (предикатом, который расположен между своими аргументами, а не перед ними).

* **Каково назначение использования алгоритма унификации?**

Алгоритм унификации необходим для попытки "увидеть одинаковость" – сопоставимость двух термов

* **Каков результат работы алгоритма унификации?**

Алгоритм унификации завершается «неудачей» или успехом. В случае успеха в результирующей ячейке сформируется подстановка.

* **В каких пределах программы переменные уникальны?**

Именованные переменные уникальны в пределах одного предложения, анонимные уникальны все.

* **Как применяется подстановка, полученная с помощью алгоритма унификации?**

Подстановка применяется к подцелям резольвенты, путем конкретизации переменных.

* **Как изменяется резольвента?**

Резольвента - текущая цель, существующая на любой стадии вычислений. Резольвенты порождаются целью и каким-либо правилом или фактом, которые просматриваются последовательно сверху вниз. Если резольвента существует при наиболее общей унификации, она вычисляется. Если пустая резольвента с помощью такой стратегии не найдена, то ответ на вопрос отрицателен.

* **В каких случаях запускается механизм отката?**

Механизм отката запускается в 2 случаях:

1. Если алгоритм попал в тупиковую ситуацию.

2. Если резольвента не пуста и решение найдено, но в базе знание остались не отмеченные предложения.

**В одной программе написать правила, позволяющие 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;

Убедиться в правильности результатов**.**

**Для каждого случая пункта 2 обосновать необходимость всех условий тела.**

**Для одного** из вариантов **ВОПРОСА** и каждого варианта **задания 2 составить таблицу**, отражающую конкретный порядок работы системы:

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

**Вопрос:…..**

|  |  |  |  |
| --- | --- | --- | --- |
| № 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 (почему и к чему приводит?) |
| 1… | … | … | Комментарий, вывод… |
| … | … | … | … |

# Практическая часть

*Листинг 1. Максимумы из 2 и 3*

**predicates**

max\_of\_two\_a(**integer**, **integer**, **integer**)

max\_of\_two\_b(**integer**, **integer**, **integer**)

max\_of\_three\_a(**integer**, **integer**, **integer**, **integer**)

max\_of\_three\_b(**integer**, **integer**, **integer**, **integer**)

**clauses**

max\_of\_two\_a(A, B, A) :- A >= B.

max\_of\_two\_a(A, B, B) :- B > A.

max\_of\_two\_b(A, B, A) :- A >= B, !.

max\_of\_two\_b(\_, B, B).

max\_of\_three\_a(A, B, C, A) :- A >= B, A >= C.

max\_of\_three\_a(A, B, C, B) :- B > A, B >= C.

max\_of\_three\_a(A, B, C, C) :- C > A, C > B.

max\_of\_three\_b(A, B, C, A) :- A >= B, A >= C, !.

max\_of\_three\_b(\_, B, C, B) :- B >= C, !.

max\_of\_three\_b(\_, \_, C, C).

**goal**

max\_of\_three\_b(3, 3, 3, Result).

**Примеры целей**

1. max\_of\_two\_a(5, 1, Result).

Result=5

1. max\_of\_two\_b(1, 4, Result).

Result=4

1. max\_of\_three\_a(2, 5, 3, Result).

Result=5

1. max\_of\_three\_b(4, 4, 4, Result).

Result=4

**Обоснование необходимости всех условий тела пункта 2**

В пункте 2.а поиск максимума реализован полным перебором, в каждом из трех правил max\_of\_three\_a два сравнения для определения является ли первое, второе или третье число максимумом соответственно. Во втором и третьем используются нестрогие неравенства, чтобы исключить вывод нескольких результатов для целей, содержащих равные числа.

В пункте 2.b поиск максимума реализован с использование предиката отсечения.

В первом проверяется, является ли первое число (А) максимумом. Если да, то значение найдено и дальнейшие проверки излишни, поэтому используется предикат отсечения.

Во втором проверяется не меньше ли второе число (В) третьего (С). Если не меньше, то оно является максимумом. А рассматривать смысла нет, т.к. уже было установлено, что оно не является максимумом, дальнейшие проверки излишни, используется предикат отсечения.

Третье является фактом, т.к. уже очевидно, что А и В не являются максимумами, значит, третье число (С) – максимум.

**Порядок работы системы** для цели max\_of\_three\_a(2, 5, 3, Result).

|  |  |  |  |
| --- | --- | --- | --- |
| № 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 (почему и к чему приводит?) |
| 1 | Цель заносится в резольвенту.  max\_of\_three\_a(2, 5, 3, Result).  Запуск процесса редукции. | Попытка унификации:  max\_of\_three\_a(2, 5, 3, Result ) = max\_of\_two\_a(A, B, A)  Результат: неудача, разные главные функторы. | Прямой ход, переход к следующему предложению. |
| 2-4 | max\_of\_three\_a(2, 5, 3, Result).  Выполнение процесса редукции. | Попытка унификации:  max\_of\_three\_a(2, 5, 3, Result).  с последующими 3-мя термами БЗ.  Результат: неудача, разные главные функторы. | Прямой ход, переход к следующему предложению. |
| 5 | max\_of\_three\_a(2, 5, 3, Result).  Выполнение процесса редукции. | Попытка унификации:  max\_of\_three\_a(2, 5, 3, Result) = max\_of\_three\_a(A, B, C, A)  Результат: успех,  подстановка: {A=2, B=5, C=3, Res=A} | Прямой ход (возможно, установка точки возврата).  Преобразование резольвенты: замена текущей цели на тело найденного с помощью унификации правила, применение полученной подстановки, |
| 6 | 2 >= 5,  2 >= 3.  Выбор верхней подцели резольвенты, запуск редукции. | 2 >= 5 – сравнение связанных переменных.  Результат: ложь. | Откат к предыдущему состоянию резольвенты, решение не найдено. |
| 7 | max\_of\_three\_a(2, 5, 3, Result).  Выполнение процесса редукции. | max\_of\_three\_a(2, 5, 3, Result) = max\_of\_three\_a(A, B, C, B)  Результат: успех,  подстановка: {A=2, B=5, C=3, Res=B} | Прямой ход (возможно, установка точки возврата).  Преобразование резольвенты: замена текущей цели на тело найденного с помощью унификации правила, применение полученной подстановки. |
| 8 | 5 > 2,  5 >= 3.  Выбор верхней подцели резольвенты, запуск редукции. | 5 > 2 – сравнение связанных переменных.  Результат: истина. | Прямой ход |
| 9 | 5 >= 3.  Выбор верхней подцели резольвенты, запуск редукции. | 5 > 2 – сравнение связанных переменных.  Результат: истина. | Решение найдено (резольвента пуста). Вывод. Выполнение отката для поиска альтернативных решений |
| 10 | max\_of\_three\_a(2, 5, 3, Result).  Выполнение процесса редукции. | Попытка унификации:  max\_of\_three\_a(2, 5, 3, Result) = max\_of\_three\_a(A, B, C, C)  Результат: успех,  подстановка: {A=2, B=5, C=3, Res=C} | Прямой ход (возможно, установка точки возврата).  Преобразование резольвенты: замена текущей цели на тело найденного с помощью унификации правила, применение полученной подстановки |
| 11 | 3 > 2,  3 > 5.  Выбор верхней подцели резольвенты, запуск редукции. | 3 > 2 – сравнение связанных переменных.  Результат: истина. | Прямой ход |
| 12 | 3 > 5.  Выбор верхней подцели резольвенты, запуск редукции. | 3 > 5 – сравнение связанных переменных.  Результат: ложь. | Откат к предыдущему состоянию резольвенты, решение не найдено. |
| 13 | max\_of\_three\_a(2, 5, 3, Result).  Выполнение процесса редукции. | Попытка унификации:  max\_of\_three\_a(2, 5, 3, Result) = max\_of\_three\_b(A, B, C, A)  Результат: неудача, разные главные функторы. | Прямой ход, переход к следующему предложению |
| 14 | max\_of\_three\_a(2, 5, 3, Result).  Выполнение процесса редукции. | Попытка унификации:  max\_of\_three\_a(2, 5, 3, Result) = max\_of\_three\_b(A, B, C, B)  Результат: неудача, разные главные функторы. | Прямой ход, переход к следующему предложению |
| 15 | max\_of\_three\_a(2, 5, 3, Result).  Выполнение процесса редукции. | Попытка унификации:  max\_of\_three\_a(2, 5, 3, Result) = max\_of\_three\_b(A, B, C, C)  Результат: неудача, разные главные функторы. | Конец БЗ.  Конец работы программы. |

**Порядок работы системы** для цели max\_of\_three\_b(2, 5, 3, Result).

|  |  |  |  |
| --- | --- | --- | --- |
| № 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 (почему и к чему приводит?) |
| 1 | Цель заносится в резольвенту.  max\_of\_three\_b(2, 5, 3, Result).  Запуск процесса редукции. | Попытка унификации:  max\_of\_three\_b(2, 5, 3, Result ) = max\_of\_two\_a(A, B, A)  Результат: неудача, разные главные функторы. | Прямой ход, переход к следующему предложению. |
| 2-7 | max\_of\_three\_a(2, 5, 3, Result).  Выполнение процесса редукции. | Попытка унификации:  max\_of\_three\_b(2, 5, 3, Result).  с последующими 6-ю термами БЗ.  Результат: неудача, разные главные функторы. | Прямой ход, переход к следующему предложению. |
| 8 | max\_of\_three\_b(2, 5, 3, Result).  Выполнение процесса редукции. | Попытка унификации:  max\_of\_three\_b(2, 5, 3, Result) = max\_of\_three\_b(A, B, C, A)  Результат: успех,  подстановка: {A=2, B=5, C=3, Res=A} | Прямой ход (возможно, установка точки возврата).  Преобразование резольвенты: замена текущей цели на тело найденного с помощью унификации правила, применение полученной подстановки, |
| 9 | 2 >= 5,  2 >= 3,  !.  Выбор верхней подцели резольвенты, запуск редукции. | 2 >= 5 – сравнение связанных переменных.  Результат: ложь. | Откат к предыдущему состоянию резольвенты, решение не найдено. |
| 10 | max\_of\_three\_b(2, 5, 3, Result).  Выполнение процесса редукции. | Попытка унификации:  max\_of\_three\_b(2, 5, 3, Result) = max\_of\_three\_b(\_, B, C, B)  Результат: успех,  подстановка: {B=5, C=3, Res=B} | Прямой ход (возможно, установка точки возврата).  Преобразование резольвенты: замена текущей цели на тело найденного с помощью унификации правила, применение полученной подстановки, |
| 11 | 5 >= 3.  !.  Выбор верхней подцели резольвенты, запуск редукции. | 5 >= 3 – сравнение связанных переменных.  Результат: истина. | Прямой ход |
| 12 | !.  Выбор верхней подцели резольвенты, запуск редукции. | Запрет отката | Решение найдено (резольвента пуста). Вывод. Завершение работы программы |