|  |  |
| --- | --- |
| 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»\_\_\_\_\_\_\_\_\_\_

**Лабораторная работа № 17**

|  |  |
| --- | --- |
| **Дисциплина** Функциональное и логическое программирование  **Тема** \_Формирование эффективных программ на Prolog\_  **Студент** \_Ильясов И. М.\_  **Группа** \_ИУ7-63Б\_  **Оценка (баллы)** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Преподаватель** \_Толпинская Н. Б., Строганов Ю. В.\_ |  |

Москва, 2020 г.

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

**Задачи работы**:

* приобрести навыки эффективного декларативного описания предметной области с использованием фактов и правил;
* изучить порядок использования фактов и правил в программе на Prolog, принципы и особенности сопоставления и отождествления термов, на основе механизма унификации. Способ формирования и изменения резольвенты. Порядок формирования ответа.

**Задание лабораторной работы**

**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… |
| … | … | … | … |

**Текст программы**

predicates

max\_with\_cut2(integer, integer, integer).

max\_without\_cut2(integer, integer, integer).

max\_with\_cut3(integer, integer, integer, integer).

max\_without\_cut3(integer, integer, integer, integer).

clauses

% Max of two numbers with using cut

max\_with\_cut2(A, B, Max) :- A > B, Max = A, !.

max\_with\_cut2(\_, B, Max) :- Max = B.

% Max of two numbers without using cut

max\_without\_cut2(A, B, Max) :- A > B, Max = A.

max\_without\_cut2(A, B, Max) :- A <= B, Max = B.

% Max of three numbers with using cut

max\_with\_cut3(A, B, C, Max) :- A > B, A > C, Max = A, !.

max\_with\_cut3(\_, B, C, Max) :- B > C, Max = B, !.

max\_with\_cut3(\_, \_, C, Max) :- Max = C.

% Max of three numbers without using cut

max\_without\_cut3(A, B, C, Max) :- A >= B, A >= C, Max = A.

max\_without\_cut3(A, B, C, Max) :- B > A, B >= C, Max = B.

max\_without\_cut3(A, B, C, Max) :- C > A, C > B, Max = C.

goal

write("Max with cut of (3, 2): "),

max\_with\_cut2(3, 2, Max);

write("Max without cut of (2, 4): "),

max\_without\_cut2(2, 4, Max);

write("\nMax with cut of (1, 2, 3): "),

max\_with\_cut3(1, 2, 3, Max);

write("Max without cut of (5, 6, 2): "),

max\_without\_cut3(5, 6, 2, Max).

**Примеры работы программы**

1. Поиск максимума из двух чисел без использования отсечения.

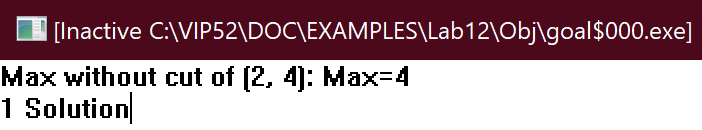


Рисунок 1. Поиск максимума из двух чисел без использования отсечения.

1. Поиск максимума из двух чисел с использованием отсечения.

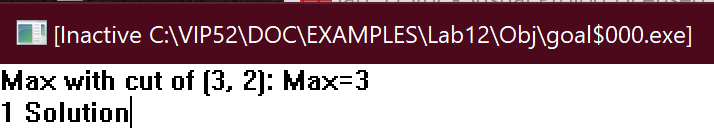


Рисунок 2. Поиск максимума из двух чисел с использованием отсечения.

1. Поиск максимума из трех чисел без использования отсечения.

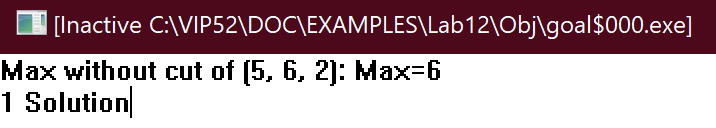


Рисунок 3. Поиск максимума из трех чисел без использования отсечения.

1. Поиск максимума из трех чисел без использования отсечения.

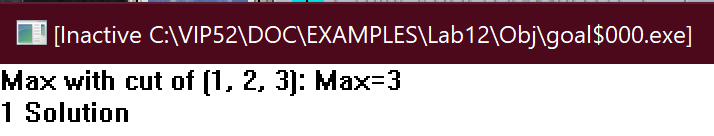


Рисунок 4. Поиск максимума из трех чисел с использованием отсечения.

**Задание с таблицей**

max\_with\_cut3(1, 2, 3, Max) – максимум 3 чисел с использованием отсечений.

|  |  |  |  |
| --- | --- | --- | --- |
| № 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 | max\_with\_cut3(1, 2, 3, Max) | T1 = max\_with\_cut3(1, 2, 3, Max).  T2 = max\_with\_cut2(A, B, Max).  Неудача, разные функторы. | Прямой ход к следующему предложению. |
| 2 | max\_with\_cut3(1, 2, 3, Max) | T1 = max\_with\_cut3(1, 2, 3, Max).  T2 = max\_with\_cut2(\_, B, Max).  Неудача, разные функторы. | Прямой ход к следующему предложению. |
| 3 | max\_with\_cut3(1, 2, 3, Max) | T1 = max\_with\_cut3(1, 2, 3, Max).  T2 = max\_without\_cut2(A, B, Max).  Неудача, разные функторы. | Прямой ход к следующему предложению. |
| 4 | max\_with\_cut3(1, 2, 3, Max) | T1 = max\_with\_cut3(1, 2, 3, Max).  T2 = max\_without\_cut2(A, B, Max).  Неудача, разные функторы. | Прямой ход к следующему предложению. |
| 5 | max\_with\_cut3(1, 2, 3, Max) | T1 = max\_with\_cut3(1, 2, 3, Max).  T2 = max\_with\_cut3(A, B, C, Max).  Успех, подстановка {A = 1, B = 2, C = 3, Max = Max} | Прямой ход к следующему предложению. |
| 6 | A > B, A > C, Max = A, !. | Проверка 1 > 2, неудача. | Прямой ход к следующему предложению. |
| 7 | max\_with\_cut3(1, 2, 3, Max) | T1 = max\_with\_cut3(1, 2, 3, Max).  T2 = max\_with\_cut3(\_, B, C, Max).  Успех, подстановка {B = 2, C = 3, Max = Max} | Прямой ход к следующему предложению. |
| 8 | B > C, Max = B, !. | Проверка 2 > 3, неудача. | Прямой ход к следующему предложению. |
| 9 | max\_with\_cut3(1, 2, 3, Max) | T1 = max\_with\_cut3(1, 2, 3, Max).  T2 = max\_with\_cut3(\_, \_, C, Max).  Успех, подстановка {C = 3, Max = Max} | Прямой ход к следующему предложению. |
| 10 | Max = С. | Max = 3. | Прямой ход. |
| 11 |  | **Результат**: Max = 3. | Обратный ход. |

max\_without\_cut3(5, 6, 2, Max) – максимум 3 чисел без использования отсечений.

|  |  |  |  |
| --- | --- | --- | --- |
| № 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 | max\_without\_cut3(5, 6, 2, Max) | T1 = max\_without\_cut3(5, 6, 2, Max).  T2 = max\_with\_cut2(A, B, Max).  Неудача, разные функторы. | Прямой ход к следующему предложению. |
| 2 | max\_without\_cut3(5, 6, 2, Max) | T1 = max\_without\_cut3(5, 6, 2, Max).  T2 = max\_with\_cut2(\_, B, Max).  Неудача, разные функторы. | Прямой ход к следующему предложению. |
| 3 | max\_without\_cut3(5, 6, 2, Max) | T1 = max\_without\_cut3(5, 6, 2, Max).  T2 = max\_without\_cut2(A, B, Max).  Неудача, разные функторы. | Прямой ход к следующему предложению. |
| 4 | max\_without\_cut3(5, 6, 2, Max) | T1 = max\_without\_cut3(5, 6, 2, Max).  T2 = max\_without\_cut2(A, B, Max).  Неудача, разные функторы. | Прямой ход к следующему предложению. |
| 5 | max\_without\_cut3(5, 6, 2, Max) | T1 = max\_without\_cut3(5, 6, 2, Max).  T2 = max\_with\_cut3(A, B, C, Max).  Неудача, разные функторы. | Прямой ход к следующему предложению. |
| 6 | max\_without\_cut3(5, 6, 2, Max) | T1 = max\_without\_cut3(5, 6, 2, Max).  T2 = max\_with\_cut3(\_, B, C, Max).  Неудача, разные функторы. | Прямой ход к следующему предложению. |
| 7 | max\_without\_cut3(5, 6, 2, Max) | T1 = max\_without\_cut3(5, 6, 2, Max).  T2 = max\_with\_cut3(\_, \_, C, Max).  Неудача, разные функторы. | Прямой ход к следующему предложению. |
| 8 | max\_without\_cut3(5, 6, 2, Max) | T1 = max\_without\_cut3(5, 6, 2, Max).  T2 = max\_without\_cut3(A, B, C, Max).  Успех, подстановка {A = 5, B = 6, C = 2, Max = Max} | Прямой ход к следующему предложению. |
| 6 | A >= B, A >= C, Max = A. | Проверка 5 > 6, неудача. | Прямой ход к следующему предложению. |
| 7 | max\_without\_cut3(5, 6, 2, Max) | T1 = max\_without\_cut3(5, 6, 2, Max).  T2 = max\_without\_cut3(A, B, C, Max).  Успех, подстановка {A = 5, B= 6, C = 3, Max = Max} | Прямой ход к следующему предложению. |
| 8 | B > A, B >= C, Max = B | Проверка 6 > 5, неудача. | Прямой ход. |
| 9 | B >= C, Max = B | Проверка 6 >= 3, неудача. | Прямой ход к следующему предложению. |
| 10 | Max = B. | Max = 6. | Прямой ход. |
| 11 |  | **Результат**: Max = 6. | Обратный ход. |

**Ответы на вопросы**

1. **Какое первое состояние резольвенты?**

Если задан простой вопрос, то сначала он попадает в резольвенту.

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

Если есть что доказывать (цель), то процесс унификации запускается автоматически. Формально: если резольвента не пуста – запускается алгоритм унификации.

1. **Каково назначение использования алгоритма унификации?**

Назначение алгоритма унификации заключается в попарном сопоставлении термов и попытке построить для них общий пример.

1. **Каков результат использования алгоритма унификации?**

Унификация может завершаться успехом или тупиковой ситуацией (неудачей).

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

Переменные уникальны в пределах предложения, т.е. в рамках предложения одно и то же имя принадлежит одной и той же переменной. Исключение – анонимные переменные (обозначаются символом нижнего подчеркивания «\_») – каждая такая переменная является отдельной сущностью и применяется, когда ее значение неважно для данного предложения.

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

Применение подстановки {*X1=T1, …, Xn=Tn*} заключается в замене каждого вхождения переменной *Xi* на соответствующий терм *Ti*.

1. **Как меняется резольвента?**

Изменение резольвенты происходит в 2 этапа:

1) из стека выбирается подцель (верхняя, т.к. стек) и для нее выполняется редукция, т.е. замена подцели на тело найденного правила;

2) к полученной конъюнкции целей применяется подстановка (наибольший общий унификатор выбранной цели и заголовка сопоставленного с этой целью правила).

1. **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.