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

ДИСЦИПЛИНА «Функциональное и логическое программирование»

**Лабораторная работа № 17**

**“Формирование эффективных программ на Prolog”**

|  |  |
| --- | --- |
| **Студент**  Чалый А. А.  **Группа** ИУ7 – 62 Б  **Оценка (баллы) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Преподаватель** Толпинская Н.Б. |  |

Москва.

2021 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 описания предметной области с использованием фактов и правил.

Изучить возможность использования системных предикатов в программе на Prolog, принципы и особенности порядка работы в этом случае. Способ формирования и изменения резольвенты в этом случае и порядок формирования ответа.

**Задание**

**Ответить на вопросы:**

* Какое первое состояние резольвенты?
* В каком случае система запускает алгоритм унификации? (т.е. Как эту необходимость на формальном уровне распознает система?)
* Каково назначение использования алгоритма унификации?
* Каков результат работы алгоритма унификации?
* В каких пределах программы переменные уникальны?
* Как применяется подстановка, полученная с помощью алгоритма унификации?
* Как изменяется резольвента?
* В каких случаях запускается механизм отката?

**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:

Листинг 1. Программа nums.pro

predicates

max(integer, integer, integer)

max\_cat(integer, integer, integer)

max(integer, integer, integer, integer)

max\_cat(integer, integer, integer)

clauses

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

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

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

max\_cat(\_, Y, Y).

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

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

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

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

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

max\_cat(\_, \_, Z, Z).

Обоснование необходимости каждого условия:

max(X, Y, Z, X):- X >= Y, X >= Z. → Необходимо проверить является ли Х наибольшим.

max(X, Y, Z, Y):- Y >= X, Y >= Z. → Необходимо проверить, что Y>X, т. к. даже если предыдущий шаг выполнен успешно, программа продолжит работу. Без этого условия, У будет ответом, например в случае Х=3 У = 2 Z= 1, что неверно.

max(X, Y, Z, Z):- Z >= X, Z >= Y. Необходимо проверить что Z больше двух других переменных, так как программа продолжит выполнение даже если предыдущие шаги выполнились успешно, что может привести к неверному результату.

в) с использованием отсечения;

max\_cat(1, 5, 3, Max) → Max = 5

Обоснование необходимости каждого условия:

max\_cat(X, Y, Z, X):- X >= Y, X >= Z, !. → Проверка что Х наибольший. Если да, то отсекаем.

max\_cat(\_, Y, Z, Y):- Y >= Z, !. → Уже проверено, что Х не максимальный, остается только проверить, что У максимальный.

max\_cat(\_, \_, Z, Z). → Х и У не максимальные → ответ Z.

Таблица 1. max(1, 5, 3, Max). (без использования отсечения)

|  |  |  |  |
| --- | --- | --- | --- |
| № 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(1, 5, 3, Max)  В качестве первого состояния в резольвенту помещается вопрос | T1= max(1, 5, 3, Max)  T2 = max(X, Y, X)  Попытка унификации  Унификация не успешна. Не совпадают арности термов | Переход к следующему предложению |
|  | max(1, 5, 3, Max) | T1= max(1, 5, 3, Max)  T2 = max(X, Y, Y)  Попытка унификации  Унификация не успешна. Не совпадают арности термов | Переход к следующему предложению |
|  | max(1, 5, 3, Max) | T1= max(1, 5, 3, Max)  T2 = max\_cat(X, Y, X)  Попытка унификации  Унификация не успешна. Разные функторы | Переход к следующему предложению |
|  | max(1, 5, 3, Max) | T1= max(1, 5, 3, Max)  T2 = max\_cat(\_, Y, Y)  Попытка унификации  Унификация не успешна. Разные функторы | Переход к следующему предложению |
|  | max(1, 5, 3, Max) | T1= max(1, 5, 3, Max)  T2 = max(X, Y, Z, X)  Попытка унификации  Унификация успешна. Подстановка:  {X=1,  Y=5,  Z=3,  Max=X} | Прямой ход. |
| 2 | 1 >= 5  1 >= 3 | 1>=5 – no | Откат |
| 3 | max(1, 5, 3, Max) | T1= max(1, 5, 3, Max)  T2 = max(X, Y, Z, Y)  Попытка унификации  Унификация успешна. Подстановка:  {X=1,  Y=5,  Z=3,  Max=Y} | Прямой ход. |
| 4 | 5 >= 1  5 >= 3 | 5 >= 1 – yes | Прямой ход |
| 5 | 5>=3 | 5>=3 -yes | Прямой ход |
| 6 | Резольвента пуста |  | Ответ найден  Откат |
| 7 | max(1, 5, 3, Max) | T1= max(1, 5, 3, Max)  T2 = max(X, Y, Z, Z)  Попытка унификации  Унификация успешна. Подстановка:  {X=1,  Y=5,  Z=3,  Max=Z} | Прямой ход. |
| 8 | 3 >= 1  3 >= 5 | 3 >=1 – yes | Прямой ход |
| 9 | 3 >= 5 | 3 >= 5 – no | Откат |
| 10 | max(1, 5, 3, Max) | T1= max(1, 5, 3, Max)  T2 = max\_cat(X, Y, Z, X)  Попытка унификации  Унификация не успешна. Разные функторы | Переход к следующему предложению |
|  | max(1, 5, 3, Max) | T1= max(1, 5, 3, Max)  T2 = max\_cat(\_, Y, Z, Y)  Попытка унификации  Унификация не успешна. Разные функторы | Переход к следующему предложению |
|  | max(1, 5, 3, Max) | T1= max(1, 5, 3, Max)  T2 = max\_cat(\_, \_, Z, Z)  Попытка унификации  Унификация не успешна. Разные функторы | Переход к следующему предложению |
|  | max(1, 5, 3, Max) | Все предложения просмотрены | Прямой ход |
| 11 | Пусто | Конец работы | Результат найден |

Таблица 2. max(1, 5, 3, Max). (с использованием отсечения)

|  |  |  |  |
| --- | --- | --- | --- |
| № 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\_cat(1, 5, 3,Max)  В качестве первого состояния в резольвенту помещается вопрос | T1= max\_cat(1, 5, 3, Max)  T2 = max(X, Y, X)  Попытка унификации  Унификация не успешна.  Функторы не совпадают | Переход к следующему предложению |
|  | max\_cat(1, 5, 3,Max) | T1= max\_cat(1, 5, 3, Max)  T2 = max(X, Y, Y)  Попытка унификации  Унификация не успешна.  Функторы не совпадают | Переход к следующему предложению |
|  | max\_cat(1, 5, 3,Max) | T1= max\_cat(1, 5, 3, Max)  T2 =max\_cat(X, Y, X)  Попытка унификации  Унификация не успешна.  Арности не совпадают | Переход к следующему предложению |
|  | max\_cat(1, 5, 3,Max) | T1= max\_cat(1, 5, 3, Max)  T2 = max\_cat(\_, Y, Y)  Попытка унификации  Унификация не успешна.  Арности не совпадают | Переход к следующему предложению |
|  | max\_cat(1, 5, 3,Max) | T1= max\_cat(1, 5, 3, Max)  T2 = max(X, Y, Z, X)  Попытка унификации  Унификация не успешна.  Арности не совпадают | Переход к следующему предложению |
|  | max\_cat(1, 5, 3,Max) | T1= max\_cat(1, 5, 3, Max)  T2 = max(X, Y, Z, Y)  Попытка унификации  Унификация не успешна.  Арности не совпадают | Переход к следующему предложению |
|  | max\_cat(1, 5, 3,Max) | T1= max\_cat(1, 5, 3, Max)  T2 = max(X, Y, Z, Z)  Попытка унификации  Унификация не успешна.  Арности не совпадают | Переход к следующему предложению |
|  | max\_cat(1, 5, 3,Max) | T1= max\_cat(1, 5, 3, Max)  T2 = max\_cat(X, Y, Z, X)  Попытка унификации  Унификация успешна.  Подстановка :  { X=1,  Y=5,  Z=3,  Max=X} | Прямой ход |
| 2 | 1 >= 5  1 >= 3  ! | 1>=5 -no | Откат |
| 3 | max\_cat(1, 5, 3,Max) | T1= max\_cat(1, 5, 3, Max)  T2 = max\_cat(\_, Y, Z, Y)  Попытка унификации  Унификация успешна.  Подстановка :  { \_=1,  Y=5,  Z=3,  Max=Y} | Прямой ход |
| 4 | 5 >= 3  ! | 5>=3 -yes | Прямой ход |
| 5 | ! | ! - указывает прологу отменить поиск альтернатив для целей до него | Прямой ход |
| 6 | Пусто | Конец | Результат найден |

**За счет чего может быть достигнута эффективность работы системы?**

**Выводы:**  Эффективность работы может быть достигнута за счет использования отсечений. Таким образом можно сократить количество проверок, выполняемых программой и соответственно сократить количество шагов, необходимых для достижения результата. Это можно наблюдать на примере таблиц. Если в первом случае выполнялись все проверки и результат был найден за 11 шагов, то во втором случае поиск результата был закончен при достижении отсечения и количество шагов составило 6.

**Контрольные вопросы:**

**Какое первое состояние резольвенты?**

Первое состояние резольвенты - заданный вопрос.

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

Процесс унификации запускается в случае, когда резольвента не пуста

**Каково назначение использования алгоритма унификации?**

Унификация позволяет формализовать процесс логического вывода. Назначение - поиск знания, которое является ответом на конкретный вопрос.

**Каков результат работы алгоритма унификации?**

Результат работы алгоритма унификации — ответ «да» или «нет», а также конкретизация переменных.

**В каких пределах программы переменные уникальны?**

Именованные переменные уникальны в рамках одного предложения, т. е. в разных предложениях одно и то же имя переменной может использоваться для обозначения разных объектов. Анонимные переменные уникальны везде.

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

Если алгоритм унификации завершился успешно и найдена подстановка, соответствующие переменные конкретизируются полученными значениями.

**Как изменяется резольвента?**

В процессе доказательства утверждений в резольвента меняется. При изменении строится новая резольвента. По стековому принципу берется верхняя подцель и заменяется на тело подходящего правила. Затем применяется найденная на текущем этапе подстановка. Успешное завершение работы программы достигается, когда резольвента пуста.

**В каких случаях запускается механизм отката?**

Механизм отката к предыдущему шагу выполняется в случае, когда унификация завершается

тупиковой ситуацией(неудачей). Кроме того, механизм используется для того, чтобы получить все возможные ответы.