



**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
Формирование эффективных программ на Prolog**

Студент Лучина Е.Д

Группа ИУ7-61Б

Преподаватель Толпинская Н.Б.

Москва.

апрель 2020 г.

Содержание отчета

[Полный текст задания.](#)

[Ответы на вопросы.](#)

[Текст программы с комментариями обозначений и идентификаторов.](#)

[Таблица, демонстрирующая работу системы при одном из успешных вариантов вопроса.](#)

[Выводы: за счет чего может быть достигнута эффективность работы системы.](#)

Задание

В одной программе написать правила, позволяющие найти

1. Максимум из двух чисел
 - a. без использования отсечения,
 - b. с использованием отсечения;
2. Максимум из трех чисел
 - a. без использования отсечения,
 - b. 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:...

Шаг	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	Комментарий, вывод...
2

Вопросы

1. Какое первое состояние резольвенты?

Вопрос, записанный в goal.

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

Резольвента не пуста, программе еще есть на какие вопросы искать ответы.

3. Каково назначение использования алгоритма унификации?

Назначение алгоритма унификации - подобрать знание, которое позволит на поставленный вопрос ответить да.

4. Каков результат работы алгоритма унификации?

Результат унификации может быть успешным (доказаны истинность утверждения) или неудачным (на основе данной базы знаний невозможно доказать утверждение). Побочным результатом унификации является конкретизация переменных.

5. В каких пределах программы переменные уникальны?

Именованные переменные уникальны в пределах предложения. Анонимные уникальны сами по себе в пределах всей программы.

6. Как применяется подстановка, полученная с помощью алгоритма унификации?

Данная подстановка применяется к существующим в резольвенте целям. Все вхождения переменных, о которых есть информация в подстановке, конкретизируются найденными значениями в дальнейших термах.

7. Как изменяется резольвента?

Преобразования резольвенты выполняются с помощью редукции - замены цели телом того правила из базы знаний, заголовок которого унифицируется с

целью. Такие правила подбираются с помощью унификации и называются сопоставимыми с целью.

Новая резольвента образуется в два этапа

1. в текущей резольвенте выбирается одна из подцелей (по стековому принципу - верхняя) и для неё выполняется редукция.
2. затем, к полученной конъюнкции целей применяется подстановка, полученная как наибольший общий унификатор цели и заголовка сопоставленного с ней правила.

Если подцель унифицируется с правилом, то число целей в резольвенте уменьшается (факт - это правило с пустым телом). Если процесс унификации зашел в тупик осуществляется откат - восстановления предыдущих резольвент и значений переменных.

8. В каких случаях запускается механизм отката?

Если алгоритм унификации зашел в тупик или найдено решение, но оно не является гарантированно последним возможным.

Листинг программы

```
domains

predicates
    max1a(integer, integer, integer)
    max1b(integer, integer, integer)

    max2a(integer, integer, integer, integer)
    max2b(integer, integer, integer, integer)

clauses

    max1a(A, B, Res):-
        A >= B, Res = A;
        A < B, Res = B.

    max1b(A, B, Res):-
        A >= B, Res = A, !;
```

Res = B.

```
max2a(A, B, C, Res) :-  
    A >= B, A >= C, Res = A;  
    B >= A, B >= C, Res = B;  
    C >= A, C >= B, Res = C.
```

```
max2b(A, B, C, Res) :-  
    A >= B, A >= C, Res = A, !;  
    B >= C, Res = B, !;  
    Res = C.
```

```
/*max2a(A, B, C, Res) :-  
    max1a(A, B, Maxab), max1a(Maxab, C, Res).
```

```
max2b(A, B, C, Res):-  
    max1b(A, B, Maxab), max1b(Maxab, C, Res).*/
```

Тестирование работы программы

з а д а н и е	П р и м е р в о п р о с а	р е з у л ь т а т
№.1 а	max1a(7, 7, Res)	Res = 7 1 Solution
	max1a(7, 9, Res)	Res = 9 1 Solution
	max1a(9, 7, Res)	Res = 9 1Solution
№.1b	max1b(5, 5, Res)	Res = 5 1 Solution
	max1b(5, 8, Res)	Res = 8 1 Solution
	max1b(8, 5, Res)	Res = 8 1Solution
№.2a	max2a(3, 3, 3, Res)	Res = 3 1 Solution

	max2a(3, 3, 2, Res)	Res = 3 1Solution
	max2a(3, 2, 3, Res)	Res = 3 1 Solution
	max2a(2, 3, 3, Res)	Res = 3 1 Solution
	max2a(4, 4, 6, Res)	Res = 6 1Solution
	max2a(4, 6, 4, Res)	Res = 6 1 Solution
	max2a(6, 4, 4, Res)	Res = 6 1 Solution
	max2a(1, 2, 3, Res)	Res = 3 1 Solution
	max2a(2, 3, 1, Res)	Res = 3 1 Solution
	max2a(3, 2, 1, Res)	Res = 3 1Solution
No.2b	max2b(3, 3, 3, Res)	Res = 3 1 Solution
	max2b(3, 3, 2, Res)	Res = 3 1Solution
	max2b(3, 2, 3, Res)	Res = 3 1 Solution
	max2b(2, 3, 3, Res)	Res = 3 1 Solution
	max2b(4, 4, 6, Res)	Res = 6 1Solution
	max2b(4, 6, 4, Res)	Res = 6 1 Solution
	max2b(6, 4, 4, Res)	Res = 6 1 Solution
	max2b(1, 2, 3, Res)	Res = 3 1 Solution

	max2b(2, 3, 1, Res)	Res = 3 1 Solution
	max2b(3, 2, 1, Res)	Res = 3 1 Solution

Также можно задать вопрос вида max1a(3, 4, 4), где последний аргумент не является переменной. Программа вернет yes если это действительно большее из представленных чисел и no иначе. Попытки поставить в вопросе переменные (именованные или анонимные) на место первых двух аргументов вызовет ошибку.

Пояснения к пункту 2

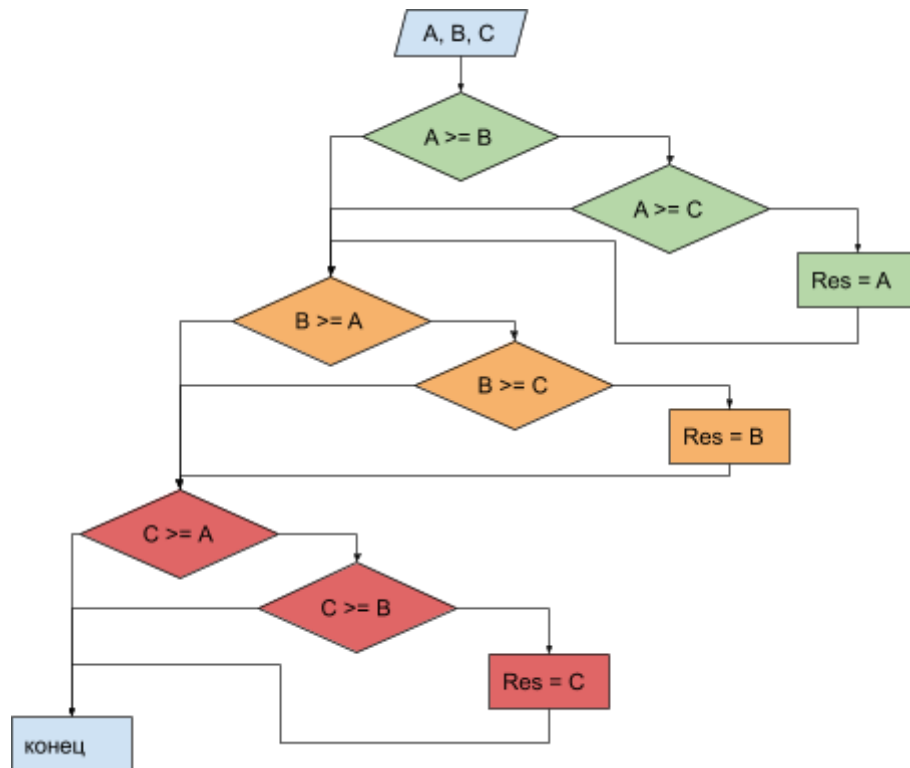
max2a(A, B, C, Res) :-

A >= B, A >= C, Res = A;

B >= A, B >= C, Res = B;

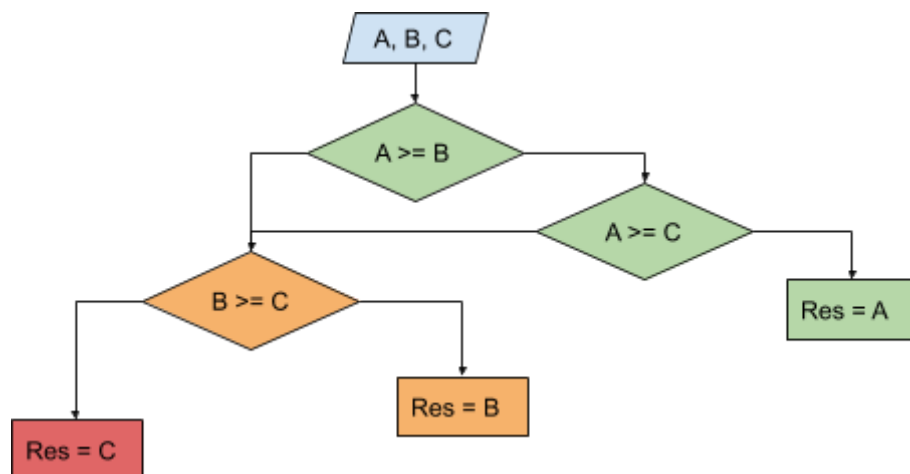
C >= A, C >= B, Res = C.

Зеленым цветом обозначена первое слагаемое дизъюнкции, рыжим - второе, красным третье. Если сравнение является истинным происходит переход по правой стрелке, иначе по левой. При получении результата, алгоритм все-равно пойдет изучать следующие строки дизъюнкции, поэтому осуществляется повторное сравнение переменных. Переход с первой строки на вторую произойдет и когда A будет меньше B, и когда A будет не меньше B и меньше C, и когда A окажется больше B и C, так что во второй части дизъюнкции мы повторно сравниваем A и B. В случае если A = B = C, Res будет связана с A, потом с B, и в конце концов и C, результатом останется C.



Для того, чтобы остановиться на найденном значении, не производить лишних сравнений и не ре-конкретизировать переменную Res, реализован второй вариант.

max2b(A, B, C, Res) :-
 A >= B, A >= C, Res = A, !;
 B >= C, Res = B, !;
 Res = C.



Было бы еще лучше после сравнения $A \geq C$ переходить сразу на C. Но программа не знает по какой причине первая часть конъюнкции оказалось ложна (потому что $A < B$ или потому что $A < C$). Она несет информацию о том

что A не является максимумом, поэтому максимальным числом является B или C , их и сравним. Если числа являются равными, то правило закончится на $Res = A$. Поставленный предикат отсечения “!” завершает правило.

Таблица

Вопрос: $\max2a(3, 5, 8, Res)$

Шаг	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	$\max2a(3, 5, 8, Res)$	Неуспешные унификации $\max2a(3, 5, 8, Res)$ с $\max1a(A, B, Res)$ и с $\max1b(A, B, Res)$	откаты
		$\max2a(3, 5, 8, Res) = \max2a(A, B, C, Res)$ Успешно $\{A = 3, B = 5, C = 8, Res = Res\}$	Прямой ход
2	$3 \geq 5,$ $3 \geq 8,$ $Res = 3;$ $5 \geq 3,$ $5 \geq 8,$ $Res = 5;$ $8 \geq 3,$ $8 \geq 5,$ $Res = 8.$	$3 \geq 5$ ложно	Первая подцель ложна, переход к следующей.
3	$5 \geq 3,$ $5 \geq 8,$ $Res = 5;$ $8 \geq 3,$ $8 \geq 5,$ $Res = 8.$	$5 \geq 3$ истинно	Эта подцель истинна

4	$5 \geq 8$, $Res = 5$; $8 \geq 3$, $8 \geq 5$, $Res = 8$.	$5 \geq 8$ ложно	Эта конъюнктивная подцель ложна, переход к следующей
5	$8 \geq 3$, $8 \geq 5$, $Res = 8$.	$8 \geq 3$ истинно	Прямой ход
6	$8 \geq 5$, $Res = 8$.	$8 \geq 5$ истинно	Прямой ход
7	$Res = 8$.	Конкретизация переменной Res $Res = 8$	Прямой ход
8	Резольвента пуста		Найдено одно решение $Res = 8$. Но существуют не отмеченные термы, нет гарантии, что решений больше нет. откат.
9	$max2a(3, 5, 8, Res)$	Неуспешна унификация $max2a(3, 5, 8, Res)$ и $max2b(A, B, C, Res)$. Больше не осталось не отмеченных термов. Завершение работы	

Вопрос: $max2b(3, 5, 8, Res)$

Шаг	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	$max2b(3, 5, 8, Res)$	Неуспешные унификации $max2b(3, 5, 8, Res)$ с каждым из следующих термов	откаты

		$\text{max1a}(A, B, \text{Res})$ $\text{max1b}(A, B, \text{Res})$ $\text{max2a}(A, B, C, \text{Res})$	
		$\text{max2b}(3, 5, 8, \text{Res}) = \text{max2b}(A, B, C, \text{Res})$ Успешно $\{A = 3, B = 5, C = 8, \text{Res} = \text{Res}\}$	Прямой ход
2	$3 \geq 5,$ $3 \geq 8,$ $\text{Res} = 3,$!; $5 \geq 8,$ $\text{Res} = 5,$!; $\text{Res} = 8.$	$3 \geq 5$ ложно	Конъюнктивная часть ложна. Переход к следующей подцели резольвенты. Прямой ход
3	$5 \geq 8,$ $\text{Res} = 5,$!; $\text{Res} = 8.$	$5 \geq 8$ ложно	Конъюнктивная часть ложна. Переход к следующей подцели резольвенты. Прямой ход
4	$\text{Res} = 8$	Конкретизация $\text{Res} = 8$	Прямой ход
5	Резольвента пуста	Достигнут конец базы знаний. Завершение работы.	

Выводы

В логических программах система выполняет полный перебор всех возможных вариантов решения, а возможности управления вычислениями – минимальны. У программиста есть право только изменить порядок следования правил в процедурах и порядок следования условий в теле правила. в Prolog включены два системных предиката: предикат отсечения и предикат fail.

Предикат отсечения ! (cut) включается в конъюнкцию целей так же, как и другие предикаты и отсекает в определенном случае бесперспективные пути доказательства. Предикат fail принудительно включает механизм отката.