



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»

Отчет по лабораторной работе №18 по курсу "Функциональное и логическое программирование"

Тема Формирование и модификация списков на Prolog

Студент Варин Д.В.

Группа ИУ7-66Б

Оценка (баллы) _____

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

Москва — 2022 г.

Условие

Используя хвостовую рекурсию, разработать эффективную программу, позволяющую:

1. Сформировать список из элементов числового списка, больших заданного значения;
2. Сформировать список из элементов, стоящих на нечетных позициях исходного списка (нумерация от 0);
3. Удалить заданный элемент из списка (один или все вхождения);
4. Преобразовать список в множество (можно использовать ранее разработанные процедуры).

Убедиться в правильности результатов.

Для одного из вариантов **ВОПРОСА** и 1-го задания **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.

№ 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?)
--------	---	--	---

Решение

Листинг 1 – Решение 1

```
1 domains
2   list = integer*.
3
4 predicates
5   f(list , integer , list).
6
7 clauses
8   f([H|T], El , [H|Res]) :- H > El , !, f(T, El , Res).
9
10  f([_|T], El , Res) :- f(T, El , Res).
11  f([], _, []) :- !.
12
13 goal
14   f([6 , 0 , 4 , 2], 3, R).
```

Листинг 2 – Решение 2

```
1 domains
2   list = integer*.
3
4 predicates
5   odd(list , list).
6
7 clauses
8   odd([_, H|T], [H|Res]) :- odd(T, Res).
9   odd([_], []) :- !.
10  odd([], []) :- !.
11
12 goal
13   odd([0 , 1 , 2 , 3 , 4 , 5 , 6], Result).
```

Листинг 3 – Решение 3 и 4

```
1 domains
2   list = integer*.
3
4 predicates
5   del(integer , list , list).
6   createSet(list , list).
7
```

```

8 clauses
9   del(EI, [EI|T], Res) :- del(EI, T, Res).
10
11   del(EI, [H|T], [H|Res]) :- del(EI, T, Res), !.
12
13   del(_, [], []).
14
15   createSet([H|T], [H|Res]) :- del(H, T, Tmp), createSet(Tmp, Res),
16   !.
17   createSet([], []).
18 goal
19   del(3, [1, 2, 3, 2, 3], Res).

```

В Таблице 1 представлен порядок поиска ответа на вопрос 1.

Таблица 1 – Порядок формирования результата для 1-го вопроса

Шаг	Сравниваемые термы; результаты	Дальнейшие действия	Результента	Подстановка
1	$f([2, 6, 4, 2], 3, R)$. и $f([H T], El, [H Res])$	Прямой ход	$2 > 3$! $f([6, 4, 2], 3, Res)$	$H = 2$ $T = [6, 4, 2]$ $El = 3$
2	$2 > 3$	Откат	! $f([6, 4, 2], 3, Res)$	$H = 2$ $T = [6, 4, 2]$ $El = 3$
3	$f([2, 6, 4, 2], 3, R)$. и $f([_ T], El, [H Res])$	Прямой ход	$f([6, 4, 2], 3, R)$	$T = [6, 4, 2]$ $El = 3$
4	$f([6, 4, 2], 3, R)$. и $f([H T], El, [H Res])$	Прямой ход	$6 > 3$! $f([4, 2], 3, Res)$	$H = 6$ $T = [4, 2]$ $El = 3$
5	$6 > 3$	Прямой ход	! $f([4, 2], 3, Res)$	$H = 6$ $T = [4, 2]$ $El = 3$
6	!	Прямой ход	$f([4, 2], 3, Res)$	$H = 6$ $T = [4, 2]$ $El = 3$
...
0	$f([, 3, [])$ и $f([, _, [])$	Прямой ход	!	$Res = [6, 4]$
Продолжение на следующей странице				

Таблица 1 – продолжение

Шаг	Сравниваемые термы; результаты	Дальнейшие действия	Резольвента	Подстановка
0	!	Завершение 1 подст. в рез-те	!	$\text{Res} = [6, 4]$
Конец таблицы				

Контрольные вопросы

Что такое рекурсия?

Рекурсия – это ссылка на описываемый объект при описании объекта.

Как организуется хвостовая рекурсия в Prolog?

- рекурсивный вызов один, расположен в конце тела правила;
- не должно быть возможности сделать откат до вычисления рекурсивного вызова.

Как организовать выход из рекурсии в Prolog?

С помощью отсечения

Какое первое состояние резольвенты?

Заданный вопрос (goal).

В каких пределах программы переменные уникальны?

Именованная переменная уникальна в предложении, в котором она используется. Анонимные переменные всегда уникальны.

В какой момент, и каким образом системе удастся получить доступ к голове списка?

Получить голову или хвост списка можно при унификации списка с $[H|T]$, H – голова списка, T – хвост списка.

Каково назначение и результат использования алгоритма унификации?

Унификация – логический вывод. Результат – подстановка.

Как формируется новое состояние резольвенты?

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:

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

Как применяется подстановка, полученная с помощью алгоритма унификации?

Подстановка применяется к целям в резольвенте путем замены текущей переменной на соответствующий терм.

В каких случаях запускается механизм отката?

Механизм отката запустится в случае неудачи алгоритма унификации.

Когда останавливается работа системы?

Работа системы останавливается, когда найдены все возможные ответы на вопрос.

Как это определяется на формальном уровне?

Когда в резольвенте находится исходный вопрос, для которого пройдена вся БЗ.