|  |  |
| --- | --- |
| **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»

**ОТЧЕТ**

*к лабораторной работе №18*

*По курсу: «Функциональное и логическое программирование»*

**Тема: «**Формирование и модификация списков на Prolog»

Студент: Зайцева А. А.

Группа: ИУ7-62Б

Преподаватели: Толпинская Н. Б.,

Строганов Ю. В.

Москва, 2022 г.

# Практическая часть

Задание. Используя хвостовую рекурсию, разработать, комментируя аргументы, 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:

|  |
| --- |
| domains  list = integer\*.  predicates  leaveGreater(list, integer, list).  leaveOddPos(list, list).  deleteOne(list, integer, list).  deleteAll(list, integer, list).  set(list, list).  clauses  leaveGreater([H|T], Num, [H|ResultTail]) :-  H > Num,  !,  leaveGreater(T, Num, ResultTail).  leaveGreater([\_|T], Num, ResultList) :-  leaveGreater(T, Num, ResultList),  !.  leaveGreater([], \_, []).  leaveOddPos([\_|[H|T]], [H|ResultList]) :-  !,  leaveOddPos(T, ResultList).  % if list has odd length  leaveOddPos([\_|[]], []) :-  !.  leaveOddPos([], []).  deleteOne([Elem|T], Elem, T) :-  !.  deleteOne([H|T], Elem, [H|ResultTail]) :-  !,  deleteOne(T, Elem, ResultTail).  deleteOne([], \_, []).  deleteAll([Elem|T], Elem, ResultList) :-  !,  deleteAll(T, Elem, ResultList).  deleteAll([H|T], Elem, [H|ResultTail]) :-  !,  deleteAll(T, Elem, ResultTail).    deleteAll([], \_, []).      set([H|T], [H|ResultTail]) :-  deleteAll(T, H, TWithoutH),  !,  set(TWithoutH, ResultTail).  set([], []).  goal  %  % leaveGreater([1, 2, -1, 3, -2], 0, Result\_).  % Result\_=[1,2,3]  % 1 Solution    %  % leaveOddPos([0, 1, 2, 3, 4, 5, 6], Result\_).  % Result\_=[1,3,5]  % 1 Solution    % leaveOddPos([0, 1, 2, 3, 4, 5], Result\_).  % Result\_=[1,3,5]  % 1 Solution      %  % deleteOne([1, 5, 1, 6, 1, 4], 1, Result\_).  % Result\_=[5,1,6,1,4]  % 1 Solution    % deleteAll([1, 5, 1, 6], 1, Result\_).  % Result\_=[5,6,4]  % 1 Solution  % set([1, 2, 3, 2, 4, 1, 6, 5, 6, 6], Result\_).  % Result\_=[1,2,3,4,6,5]  % 1 Solution  set([1, 2, 1], Result\_).  % Result\_=[1,2]  % 1 Solution |

Вопрос: set([1, 2, 1], Result).

|  |  |  |  |
| --- | --- | --- | --- |
| № шага | Текущая резольвента–ТР | ТЦ, выбираемые правила: сравниваемые термы, подстановка | Дальнейшие действия с  комментариями |
| 0 | set([1, 2, 1], Result\_) |  | Запуск алгоритма унификации для вопроса, с начала БЗ |
| 1 | set([1, 2, 1], Result\_) | set([1, 2, 1], Result\_)=  leaveGreater([H|T], Num, [H|ResultTail])  Сравнение главных функторов:  oddSum = lengthInner  Унификация неуспешна | Резольвента не менятеся (неуспешная унификацияя с заголовком)  Прямой ход, переход к следующему предложению БЗ |
| … |  |  |  |
| 13 | set([1, 2, 1], Result\_) | set([1, 2, 1], Result\_)=  set([H|T], [H|ResultTail])  Унификация успешна  Подстановка: {H=1, T=[2, 1], Result\_=[H|ResultTail]} | Верхняя подцель успешно унифицировалась с заголовком правила. Образование новой резольвенты:  1. Редукция верхней подцели - замена телом найденного правила:  deleteAll(T, H, TWithoutH),  !,  set(TWithoutH, ResultTail).  2. Применение подстановки к полученной  конъюнкции целей. Новое состояние резольвенты:  deleteAll([2, 1], 1, TWithoutH),  !,  set(TWithoutH, ResultTail).  Резольвента непуста, запуск алгоритма унификации для верхней подцели c начала БЗ |
| … |  |  |  |
| 23 | deleteAll([2, 1], 1, TWithoutH),  !,  set(TWithoutH, ResultTail). | deleteAll([2, 1], 1, TWithoutH)=  deleteAll([Elem|T], Elem, ResultList)  Унификация неуспешна | Прямой ход, переход к следующему предложению БЗ |
| 24 | deleteAll([2, 1], 1, TWithoutH),  !,  set(TWithoutH, ResultTail). | deleteAll([2, 1], 1, TWithoutH)=  deleteAll([H|T], Elem, [H|ResultTail2]) % 2 введено, чтобы не запутаться  Унификация успешна  Подстановка: {H=2, T=[1], Elem=1, TWithoutH=[H|ResultTail2]} | Верхняя подцель успешно унифицировалась с заголовком правила. Образование новой резольвенты:  1. Редукция верхней подцели: замена телом найденного правила:  !,  deleteAll(T, Elem, ResultTail2),  !,  set(TWithoutH, ResultTail).  2. Применение подстановки к полученной  конъюнкции целей. Новое состояние резольвенты:  !,  deleteAll([1], 1, ResultTail2),  !,  set(TWithoutH, ResultTail).  Резольвента непуста, запуск алгоритма унификации для верхней подцели |
| 25 | !,  deleteAll([1], 1, ResultTail2),  !,  set(TWithout, ResultTail). | Унификация успешна | Встречен системный предикат отсечения, который на обратном ходе запрещает использование других правил процдуры deleteAll. Редукция верхней подцели |
| … |  |  |  |
| 34 | deleteAll([1], 1, ResultTail2),  !,  set(TWithoutH, ResultTail). | deleteAll([1], 1, ResultTail2)=  deleteAll([Elem|T], Elem, ResultList)  Унификация успешна  Подстановка: {T=[], Elem=1, ResultTail2=ResultList} | Верхняя подцель успешно унифицировалась с заголовком правила. Образование новой резольвенты:  1. Редукция верхней подцели: замена телом найденного правила:  !,  deleteAll(T, Elem, ResultList),  !,  set(TWithoutH, ResultTail).  2. Применение подстановки к полученной  конъюнкции целей. Новое состояние резольвенты:  !,  deleteAll([], 1, ResultTail2),  !,  set(TWithoutH, ResultTail).  Резольвента непуста, запуск алгоритма унификации для верхней подцели |
| … |  |  |  |
| 47 | deleteAll([], 1, ResultTail2),  !,  set(TWithoutH, ResultTail). | deleteAll([], 1, ResultTail2)=  deleteOne([], \_, [])  Унификация успешна  Подстановка: { ResultTai2=[]}  Также появлляется значение (с шага 24) TWithoutH=[H|ResultTail2]=[2] | Верхняя подцель успешно унифицировалась с заголовком правила. Образование новой резольвенты:  1. Редукция верхней подцели: удаление deleteAll([], 1, ResultTail2)  2. Применение подстановки к полученной  конъюнкции целей. Новое состояние резольвенты:  !,  set([2], ResultTail).  Резольвента непуста, запуск алгоритма унификации для верхней подцели |
| … |  |  |  |
| 61 | set([2], ResultTail) | set([2], ResultTail)=  set([H|T], [H|ResultTail2]) % 2 введено, чтобы не запутаться  Унификация успешна  Подстановка: {H=2, T=[], ResultTail=[H|ResultTail2]} | Верхняя подцель успешно унифицировалась с заголовком правила. Образование новой резольвенты:  1. Редукция верхней подцели: замена телом найденного правила:  deleteAll(T, H, TWithoutH),  !,  set(TWithoutH, ResultTail).  2. Применение подстановки к полученной  конъюнкции целей. Новое состояние резольвенты:  deleteAll([], 2, TWithoutH),  !,  set(TWithoutH, ResultTail2).  Резольвента непуста, запуск алгоритма унификации для верхней подцели |
| … |  |  |  |

