



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 №11
по дисциплине «Функциональное и логическое
программирование»**

Тема Рекурсия на Prolog

Студент Золотухин А.В.

Группа ИУ7-64Б

Оценка (баллы) _____

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

Задание

Используя хвостовую рекурсию, разработать (комментируя назначение аргументов) эффективную программу, позволяющую:

1. Найти длину списка (по верхнему уровню);
2. Найти сумму элементов числового списка;
3. Найти сумму элементов числового списка, стоящих на нечетных позициях исходного списка (нумерация от 0);
4. Сформировать список из элементов числового списка, больших заданного значения;
5. Удалить заданный элемент из списка (один или все вхождения);
6. Объединить два списка.

Решение

1. Найти длину списка (по верхнему уровню)

```
1 domains
2   list = integer*
3   n = integer
4
5 predicates
6   len(list, n).
7   lenHelp(list, n, n).
8
9 clauses
10  len(L, Res) :- lenHelp(L, Res, 0).
11  lenHelp([], Res, Res) :- !.
12  lenHelp([_|T], Res, CurLen) :- NewCurLen = CurLen + 1,
13                                lenHelp(T, Res, NewCurLen).
14
15 goal
16  len([1, 2, 3], Res).
```

2. Найти сумму элементов числового списка

```
1 domains
2   list = integer*
3   n = integer
4
5 predicates
6   sum(list, n).
7   sumHelp(list, n, n).
8
9 clauses
10  sum(L, Res) :- sumHelp(L, Res, 0).
11  sumHelp([], Res, Res) :- !.
12  sumHelp([H|T], Res, CurSum) :- NewCurSum = CurSum + H,
13                                sumHelp(T, Res, NewCurSum).
14
15 goal
16  sum([1, 2, 3], Res).
```

3. Найти сумму элементов числового списка, стоящих на нечетных позициях исходного списка

```
1 domains
2   list = integer*
3   n = integer
4
5 predicates
6   sumOddPos(list, n).
7   sumHelp(list, n, n).
8
9 clauses
10  sumOddPos(L, Res) :- sumHelp(L, Res, 0).
11  sumHelp([], Res, Res) :- !.
12  sumHelp([_|[]], Res, Res) :- !.
13  sumHelp([_|[H|T]], Res, CurSum) :- NewCurSum = CurSum + H,
14  sumHelp(T, Res, NewCurSum).
15
16 goal
17  sumOddPos([1, 2, 3], Res).
```

4. Сформировать список из элементов числового списка, больших заданного значения

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

5. Удалить заданный элемент из списка (один или все вхождения)

```
1 domains
2   list = integer*
3   n = integer
4
5 predicates
6   removeAll(list, n, list).
7   removeOne(list, n, list).
8
9 clauses
10  removeAll([N|T], N, ResT) :- removeAll(T, N, ResT), !.
11  removeAll([H|T], N, [H|Res]) :- removeAll(T, N, Res).
12  removeAll([], _, []).
13
14  removeOne([N|T], N, T) :- !.
15  removeOne([H|T], N, [H|ResT]) :- removeOne(T, N, ResT), !.
16  removeOne([], _, []).
17
18 goal
19  removeAll([1, 2, 3, 2], 2, Res).
20  %removeOne([1, 2, 3, 2], 2, Res).
```

6. Объединить два списка

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

Вопрос: $\text{append}([1, 2], [3, 4], \text{Res})$.

№ 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 (к чему приводит?)
0	$\text{append}([1, 2], [3, 4], \text{Res})$. Резольвента непуста, запуск алгоритма унификации для верхней подцели, с начала БЗ		Прямой ход. Запуск алгоритма унификации для верхней подцели, с начала БЗ
1	$\text{append}([1, 2], [3, 4], \text{Res})$. Резольвента не меняется (неуспешная унификация с заголовком)	$\text{append}([1, 2], [3, 4], \text{Res}) = \text{append}([], L, L)$ Сравнение главных функторов: $\text{append} = \text{append}$ Сравнение аргументов: $[1, 2] = [], [3, 4] = L, \text{Res} = L$ Унификация неуспешна.	Прямой ход, переход к следующему предложению БЗ
2	$\text{append}([1, 2], [3, 4], \text{Res})$. Верхняя подцель успешно унифицировалась с заголовком правила. Образование новой резольвенты: 1. Редукция верхней подцели: замена $\text{append}([1, 2], [3, 4], \text{Res})$ телом найденного правила: $\text{append}(T, L, \text{ResT})$. 2. Применение подстановки к полученной конъюнкции целей. Новое состояние резольвенты: $\text{append}([2], [3, 4], \text{ResT})$. Резольвента непуста, запуск алгоритма унификации для верхней подцели	$\text{append}([1, 2], [3, 4], \text{Res}) = \text{append}([H T], L, [H \text{ResT}])$ Сравнение главных функторов: $\text{append} = \text{append}$ Сравнение аргументов: $[1, 2] = [H T], [3, 4] = L, \text{Res} = [H \text{ResT}]$ Унификация успешна Подстановка: $\{H=1, T=[2], L = [3, 4], \text{Res}=[1 \text{ResT}]\}$	Прямой ход. Новое состояние резольвенты
3
4	$\text{append}([2], [3, 4], \text{ResT})$. 1. Редукция верхней подцели: замена $\text{append}([2], [3, 4], \text{ResT})$ телом найденного правила: $\text{append}(T, L, \text{ResT})$. 2. Применение подстановки к полученной конъюнкции целей. Новое состояние резольвенты: $\text{append}([], [3, 4], \text{ResT})$. Резольвента непуста, запуск алгоритма унификации для верхней подцели	$\text{append}([2], [3, 4], \text{ResT}) = \text{append}([H T], L, [H \text{ResT}])$ Сравнение главных функторов: $\text{append} = \text{append}$ Сравнение аргументов: $[2] = [H T], [3, 4] = L, \text{ResT} = [H \text{ResT}]$ Унификация успешна Подстановка: $\{H=2, T=[], L = [3, 4], \text{ResT}=[2 \text{ResT}]\}$	Прямой ход. Новое состояние резольвенты
5	$\text{append}([], [3, 4], \text{ResT})$. 1. Редукция верхней подцели: замена $\text{append}([], [3, 4], \text{ResT})$ телом найденного правила: !. 2. Применение подстановки к полученной конъюнкции целей. Новое состояние резольвенты: !. Резольвента непуста, запуск алгоритма унификации для верхней подцели	$\text{append}([], [3, 4], \text{ResT}) = \text{append}([], L, L)$ Сравнение главных функторов: $\text{append} = \text{append}$ Сравнение аргументов: $[] = [], [3, 4] = L, \text{ResT} = L$ Унификация успешна Подстановка: $\{L=[3, 4], \text{ResT}=[3, 4]\}$	Прямой ход. Новое состояние резольвенты
7	! Отмена последней редукции, восстановление предыдущего состояния резольвенты: $\text{append}([1, 2], [3, 4], \text{Res})$.		Встречен системный предикат отсечения. Решение найдено: формируется подстановка $\{\text{Res}=[1, 2, 3, 4]\}$ в качестве побочного эффекта. Завершение работы.