



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 № 17

по курсу «Функциональное и логическое программирование»

на тему: «Обработка списков на Prolog»

Студент ИУ7-61Б
(Группа)

(Подпись, дата)

Мицевич М. Д.
(И. О. Фамилия)

Преподаватель

(Подпись, дата)

Толшинская Н. Б.
(И. О. Фамилия)

2022 г.

Постановка задачи

Задание: используя хвостовую рекурсию, разработать эффективную программу (комментируя назначение аргументов), 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);

Листинг 1 – Решение задания №1

```
domains
    list = integer*

predicates
    length_tmp(integer, integer, list)
    length(integer, list)

    sum_tmp(integer, integer, list)
    sum(integer, list)

    oddsum_tmp(integer, integer, integer, list)
    oddsum(integer, list)

clauses
    length_tmp(Res, Len, [_ | Tail]) :- NewLen = Len + 1,
        length_tmp(Res, NewLen, Tail).
    length_tmp(Res, Len, []) :- Res = Len, !.
    length(Res, List) :- length_tmp(Res, 0, List).

    sum_tmp(Res, Sum, [Head | Tail]) :- NewSum = Sum + Head,
        sum_tmp(Res, NewSum, Tail).
    sum_tmp(Res, Sum, []) :- Res = Sum, !.
    sum(Res, List) :- sum_tmp(Res, 0, List).

    oddsum_tmp(N, Res, Sum, [Head | Tail]) :- N mod 2 = 1, N1
        = N + 1, NewSum = Sum + Head, oddsum_tmp(N1, Res,
            NewSum, Tail), !.
    oddsum_tmp(N, Res, Sum, [Head | Tail]) :- N1 = N + 1,
        oddsum_tmp(N1, Res, Sum, Tail), !.
```

```
oddsum_tmp(N, Res, Sum, []) :- Res = Sum, !.  
oddsum(Res, List) :- oddsum_tmp(0, Res, 0, List).  
  
goal  
  
%length(Res, [1, 2, 3, 4]).  
%sum(Res, [1, 2, 3, 4]).  
oddsum(Res, [1, 2, 3, 4]).
```

Рисунок 1 – Таблица к заданию.

Но 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?)
1	$\text{length}(\text{Res}, [1, 2])$	$\text{length_tmp}(\text{Res}, \text{Len}, [_ \text{Tail}])$ и $\text{length}(\text{Res}, [1, 2])$ Результат: неудача, термы не унифицируемы	Прямой ход. Переход к следующему предложению.
2	/--/	/--/	/--/
3	$\text{length}(\text{Res}, [1, 2])$	$\text{length}(\text{Res}, [1, 2])$ и $\text{length}(\text{Res}, \text{List})$ Результат: успех + подстановка $\{\text{List} = [1, 2], \text{Res} = \text{Res}\}$	Заголовок правила заменяется его телом с учетом подстановки
4	$\text{length_tmp}(\text{Res}, 0, [1, 2]), !$	$\text{length_tmp}(\text{Res}, \text{Len}, [_ \text{Tail}])$ и $\text{length_tmp}(\text{Res}, 0, [1, 2]),$ Результат: успех + подстановка $\{\text{Len} = 0, \text{Res} = \text{Res}, \text{Tail} = [2]\}$	Заголовок правила заменяется его телом с учетом подстановки
5	$\text{NewLen} = 0 + 1,$ $\text{length_tmp}(\text{Res}, \text{NewLen}, [2]), !$	$\text{NewLen} = 0 + 1$ Результат: успех	Переход к следующему терму
6	$\text{length_tmp}(\text{Res}, 1, [2]), !$	$\text{length_tmp}(\text{Res}, 1, [2])$ и $\text{length_tmp}(\text{Res}, \text{Len}, [_ \text{Tail}])$ Результат: успех + подстановка $\{\text{Len} = 1, \text{Res} = \text{Res}, \text{Tail} = []\}$	Заголовок правила заменяется его телом с учетом подстановки:
7	$\text{NewLen} = 1 + 1,$ $\text{length_tmp}(\text{Res}, \text{NewLen}, []), !$	$\text{NewLen} = 1 + 1$ Результат: успех	Переход к следующему терму
8	$\text{length_tmp}(\text{Res}, 2, []), !$	$\text{length_tmp}(\text{Res}, \text{Len}, [_ \text{Tail}])$ и $\text{length_tmp}(\text{Res}, 2, [])$ Результат: неудача, термы не унифицируемы	Прямой ход. Переход к следующему предложению.
9	$\text{length_tmp}(\text{Res}, 2, []), !$	$\text{length_tmp}(\text{Res}, 2, [])$ и $\text{length_tmp}(\text{Res}, \text{Len}, [])$ Результат: успех + подстановка $\{\text{Len} = 2, \text{Res} = \text{Res}\}$	Заголовок правила заменяется его телом с учетом подстановки
10	$\text{Res} = 2, !, !$	$\text{Res} = 2$ Результат: успех	Переход к следующему терму
11			Найдено решение $\text{Res}=2$ Отсечение \Rightarrow система больше не будет искать решение в процедуре length_tmp