



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»

НАПРАВЛЕНИЕ ПОДГОТОВКИ \_\_\_\_\_ «09.03.04 Программная инженерия»

## ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №11

Название: \_\_\_\_\_ Рекурсии

Дисциплина: \_\_\_\_\_ Функциональное и логическое программирование

Студент	ИУ7-66Б	_____	А.Д. Ковель
	Группа	Подпись, дата	И. О. Фамилия

Преподаватель	_____	Н. Б. Толпинская
---------------	-------	------------------

Преподаватель	_____	Ю. В. Строганов
	Подпись, дата	И. О. Фамилия

Москва, 2023 г.

# 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.

```
1 list_length(List, Length) :- list_length(List, 0, Length).
2 list_length([], Acc, Acc) :- !.
3 list_length([_|T], Acc, Length) :- !,
4     NewAcc is Acc + 1,
5     list_length(T, NewAcc, Length).
6
7 sum_list(List, Sum) :- sum_list(List, 0, Sum), !.
8 sum_list([], Acc, Acc) :- !.
9 sum_list([H|T], Acc, Sum) :-
10     NewAcc is Acc + H,
11     sum_list(T, NewAcc, Sum).
12
13 sum_even_positions(List, Sum) :- !,
14     sum_even_positions(List, 0, Sum).
15
16 sum_even_positions([], Sum, Sum) :- !.
17 sum_even_positions([_], Sum, Sum).
18 sum_even_positions([_, X|T], Acc, Sum) :-
19     NewAcc is Acc + X,
20     sum_even_positions(T, NewAcc, Sum).
21
```

```

22 filter_gt_y([], _, []) :- !.
23 filter_gt_y([X|Xs], Y, [X|Ys]) :- X > Y, !, filter_gt_y(Xs, Y,
    Ys).
24 filter_gt_y([X|Xs], Y, Ys) :- X ≤ Y, !, filter_gt_y(Xs, Y, Ys).
25
26 delete_elem_once(_, [], []) :- !.
27 delete_elem_once(X, [X|Tail], Tail) :- !.
28 delete_elem_once(X, [Y|Tail], [Y|Tail1]) :-
29     delete_elem_once(X, Tail, Tail1).
30
31 delete_all(_, [], []) :- !.
32 delete_all(X, [X|Tail], Result) :- !,
33     delete_all(X, Tail, Result).
34 delete_all(X, [Y|Tail], [Y|Tail1]) :- delete_all(X, Tail,
    Tail1).
35
36 concat([], L, L) :- !.
37 concat([H|T], L, [H|Result]) :- concat(T, L, Result).

```

Таблица 0.1 – Порядок работы системы для  $\text{list\_length}([1,2,3], \text{Res})$

N	Состояние резольвенты	Для каких термов запускается алгоритм унификации и результат подстановки	Дальнейшие действия
1	$\text{list\_length}([1,2,3], \text{Res})$ .	$\text{list\_length}([1,2,3], 0)$ . Результат: Унификация прошла успешно, переменная Асс принимает значение 0.	Прямой ход.
2	$\text{list\_length}([1,2,3], \text{Acc}, \text{Res})$ , $\text{Res} = \text{Length}$ , $\text{Acc} = 0$ .	$\text{list\_length}([1,2,3], 0)$ . Результат: Унификация прошла успешно, переменная Асс принимает значение 0.	Прямой ход.
2	$\text{list\_length}([2,3], 1, \text{Res})$ .	$\text{list\_length}([2,3], 1, \text{Length})$ . Результат: Унификация прошла успешно, переменная Асс принимает значение 1.	Прямой ход.
3	$\text{list\_length}(\text{List}, \text{Acc}, \text{Res})$ , $\text{Res}=0$ , $\text{Acc}=1$ , $\text{List}=[2,3]$ .	$\text{list\_lengthh}(\text{List}, \text{Acc}, \text{Res})$ $\text{list\_lengthth}([\_ \text{T}], \text{Acc}, \text{Length})$ . Результат: Унификация прошла успешно, переменная NewAcc принимает значение $\text{Acc} + 1 = 2$ , $\text{List}=\text{T}$ .	Прямой ход.
4	$\text{list\_length}(\text{List}, \text{Acc}, \text{Res})$ , $\text{Res}=0$ , $\text{Acc}=2$ , $\text{List}=[3]$ .	$\text{list\_lengthh}(\text{List}, \text{Acc}, \text{Res})$ $\text{list\_lengthth}([\_ \text{T}], \text{Acc}, \text{Length})$ . Результат: Унификация прошла успешно, переменная NewAcc принимает значение $\text{Acc} + 1 = 3$ , $\text{List}=\text{T}$ .	Прямой ход.
5	$\text{list\_length}(\text{List}, \text{Acc}, \text{Res})$ , $\text{Res}=0$ , $\text{Acc}=3$ , $\text{List}=[]$	$\text{list\_lengthh}([], \text{Acc}, \text{Acc})$ . Результат: Унификация прошла успешно, переменная Асс принимает значение 3.	Обратный ход.
6	$\text{list\_length}(\text{List}, \text{Acc}, \text{Res})$ , $\text{Res}=3$ , $\text{Acc}=3$ , $\text{List}=[]$	$\text{list\_lengthh}([], \text{Acc}, \text{Acc})$ . Результат: Унификация прошла успешно.	Обратный ход.
7	$\text{list\_length}([1,2,3], \text{Res})$ .	$\text{list\_lengthh}([1,2,3], 0)$ . Результат: Унификация прошла успешно.	Обратный ход.