



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 Завойских Е.В.

Группа ИУ7-63Б

Оценка (баллы) \_\_\_\_\_

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

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 domains
2   list = integer*.
3   num = integer.
4
5 predicates
6   len_r(list , num, num).
7   len(list , num).
8
9   sum_r(list , num, num).
10  sum(list , num).
11
12  sum2_r(list , num, num).
13  sum2(list , num).
14
15  els_large_r(list , list , num, list).
16  els_large(list , list , num).
17
18  del_el_r(list , list , num, list).
19  del_el(list , list , num).
20
21  merge(list , list , list).
22
23
24 clauses
25   % 1
26   len_r([], Res, Res) :- !.
27   len_r([_|T], Res, Cur) :- Cur2 = Cur + 1,
28                             len_r(T, Res, Cur2).
29
30   len(L, Res) :- len_r(L, Res, 0).
31
32   %2
33   sum_r([], Res, Res) :- !.
```

```

34 sum_r([H|T], Res, Cur) :- Cur2 = Cur + H,
35     sum_r(T, Res, Cur2).
36
37 sum(L, Res) :- sum_r(L, Res, 0).
38
39 %3
40 sum2_r([], Res, Res) :- !.
41
42 sum2_r([_], Res, Res) :- !.
43
44 sum2_r([_, H|T], Res, Cur) :- Cur2 = Cur + H,
45     sum2_r(T, Res, Cur2).
46
47 sum2(L, Res) :- sum2_r(L, Res, 0).
48
49 %4
50 els_large_r([], Res, _, Res) :- !.
51
52 els_large_r([H|T], Res, N, Cur) :- H > N, !,
53     els_large_r(T, Res, N, [H|Cur]).
54
55 els_large_r([_|T], Res, N, Cur) :- els_large_r(T, Res, N, Cur).
56
57
58 els_large(L, Res, N) :- els_large_r(L, Res, N, []).
59
60 %5
61 del_el_r([], Res, _, Res) :- !.
62
63 del_el_r([H|T], Res, N, Cur) :- H <> N, !,
64     del_el_r(T, Res, N, [H|Cur]).
65
66 del_el_r([_|T], Res, N, Cur) :- del_el_r(T, Res, N, Cur).
67
68
69 del_el(L, Res, N) :- del_el_r(L, Res, N, []).
70
71 %6
72 merge([], L2, L2) :- !.
73 merge([H|T], L2, [H|T3]) :- merge(T, L2, T3).
74
75 goal
76 %len([1, 2, 3, 4, 5], Res).
77 %sum([1, 2, 3, 4, 5], Res).
78 %sum2([1, 2, 3, 4, 5], Res).
79 %els_large([3, 4, 5, 1, 0], Res, 2).
80 %del_el([3, 4, 5, 1, 0, 4], Res, 4).
81 merge([3, 4, 5], [1, 0], Res).

```