



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 ИУ7-63Б  
(Группа)

\_\_\_\_\_  
(Подпись, дата)

Миронов Г. А.  
(И. О. Фамилия)

Преподаватель

\_\_\_\_\_  
(Подпись, дата)

Толшинская Н. Б.  
(И. О. Фамилия)

2022 г.

# 1 Практическая часть

**Задание 17:** Используя хвостовую рекурсию, разработать эффективную программу, (комментируя назначение аргументов), позволяющую:

- найти длину списка (по верхнему уровню);
- найти сумму элементов числового списка;
- найти сумму элементов числового списка, стоящих на нечетных позициях исходного списка (нумерация от 0).

Убедиться в правильности результатов.

Для одного из вариантов **ВОПРОСА** и одного из заданий составить таблицу, отражающую конкретный порядок работы системы.

Т.к. резольвента хранится в виде стека, то состояние резольвенты требуется отображать в столбик: вершина – сверху! Новый шаг надо начинать с нового состояния резольвенты! Для каждого запуска алгоритма унификации, требуется указать № выбранного правила и дальнейшие действия – и почему.

## Листинг 1.1 – Реализация вычисления числа Фибоначчи

```

1 domains
2     list = integer*.
3
4 predicates
5     len(list, integer).
6     len(list, integer, integer).
7     sum(list, integer).
8     sum(list, integer, integer).
9     sumOddPos(list, integer).
10    sumOddPos(list, integer, integer).
11
12 clauses
13     len(List, Len) :- len(List, 0, Len).
14     len([], Len, Len) :- !.
15     len([_|T], CurLen, Len) :-
16         NewLen = CurLen + 1,
17         len(T, NewLen, Len).
18
19     sum(List, Sum) :- sum(List, 0, Sum).
20     sum([], Sum, Sum) :- !.
21     sum([H|T], CurSum, Sum) :-
22         NewSum = CurSum + H,
23         sum(T, NewSum, Sum).
24
25     sumOddPos(List, Sum) :- sumOddPos(List, 0, Sum).
26     sumOddPos([], Sum, Sum) :- !.
27     sumOddPos([_|_], Sum, Sum) :- !.
28     sumOddPos([_|[H|T]], CurSum, Sum) :-
29         NewSum = CurSum + H,
30         sumOddPos(T, NewSum, Sum).
31
32 goal
33     %len([0, 1, -2, 10], Len).
34     %sum([0, -2, 10], Sum).
35     %sumOddPos([1, 2, 1, 2, 1], Sum).
36     %sumOddPos([1, 2, 1, 2], Sum).

```

В Таблице 1.1 представлен порядок поиска ответа на вопрос 1.

Таблица 1.1 – Порядок формирования результата для 1-го вопроса

Шаг	Сравниваемые термы; результаты	Дальнейшие действия	Резольвента	Подстановка
1	$\text{len}([0, 1, -2, 10], \text{Len})$ и $\text{len}(\text{List}, \text{Len})$	Прямой ход	$\text{len}([0, 1, -2, 10], 0, \text{Len})$	$\text{List} = [0, 1, -2, 10]$
2	$\text{len}([0, 1, -2, 10], 0, \text{Len})$ и $\text{len}(\text{List}, \text{Len})$ Не унифицируемы	Прямой ход Переход к след. предл.	$\text{len}([0, 1, -2, 10], 0, \text{Len})$	$\text{List} = [0, 1, -2, 10]$
...	...	...	...	...
4	$\text{len}([0, 1, -2, 10], 0, \text{Len})$ и $\text{len}([\_ \text{T} ], \text{CurLen}, \text{Len})$	Прямой ход	$\text{NewLen} = \text{CurLen} + 1$ $\text{len}([1, -2, 10], \text{NewLen}, \text{Len})$	$\text{T} = [1, -2, 10]$ $\text{CurLen} = 0$
5	$\text{NewLen} = \text{CurLen} + 1$	Прямой ход	$\text{len}([1, -2, 10], 1, \text{Len})$	$\text{T} = [1, -2, 10]$ $\text{CurLen} = 0$ $\text{NewLen} = 1$
...	...	...	...	...
8	$\text{len}([1, -2, 10], 1, \text{Len})$ и $\text{len}([\_ \text{T} ], \text{CurLen}, \text{Len})$	Прямой ход	$\text{NewLen} = \text{CurLen} + 1$ $\text{len}([-2, 10], \text{NewLen}, \text{Len})$	$\text{T} = [-2, 10]$ $\text{CurLen} = 1$
Продолжение на следующей странице				

Таблица 1.1 – продолжение

Шаг	Сравниваемые термы; результаты	Дальнейшие действия	Резольвента	Подстановка
9	$\text{NewLen} = \text{CurLen} + 1$	Прямой ход	$\text{len}([-2, 10], \text{NewLen}, \text{Len})$	$T = [-2, 10]$ $\text{NewLen} = 2$
...	...	...	...	...
19	$\text{len}([], 4, \text{Len})$ и $\text{len}([], \text{Len}, \text{Len})$	прямой ход	!	$T = []$ $\text{NewLen} = 4$ $\text{Len} = 4$
20	!	Завершение работы 1 подст. в рез-те		$\text{Len} = 4$
Конец таблицы				

## **2 Контрольный вопросы**

### **2.1 Что такое рекурсия?**

Рекурсия – это ссылка на описываемый объект при описании объекта.

### **2.2 Как организуется хвостовая рекурсия в Prolog?**

- рекурсивный вызов единственен и расположен в конце тела правила;
- не должно быть возможности сделать откат до вычисления рекурсивного вызова.

### **2.3 Как организовать выход из рекурсии в Prolog?**

С помощью отсечения

### **2.4 Какое первое состояние резольвенты?**

Заданный вопрос (goal).

### **2.5 В каких пределах программы переменные уникальны?**

Именованная переменная уникальна в рамках предложения, в котором она используется. Анонимные переменные всегда уникальны.

### **2.6 В какой момент, и каким способом системе удастся получить доступ к голове списка?**

Получить голову или хвост списка можно при унификации списка с  $[H|T]$ ,  $H$  – голова списка,  $T$  – хвост списка.

### **2.7 Каково назначение и результат использования алгоритма унификации?**

Унификация – механизм логического вывода. Результат – подстановка.

## **2.8 Как формируется новое состояние резольвенты?**

Преобразования резольвенты выполняются с помощью редукции. Редукцией цели  $G$  с помощью программы  $P$  называется замена цели  $G$  телом того правила из  $P$ , заголовок которого унифицируется с целью. Новая резольвента образуется в два этапа:

- в текущей резольвенте выбирается одна из подцелей и для неё выполняется редукция;
- к полученной конъюнкции целей применяется подстановка, полученная как наибольший общий унификатор цели и заголовка сопоставленного с ней правила.

## **2.9 Как применяется подстановка, полученная с помощью алгоритма унификации? Как глубоко?**

Подстановка применяется к целям в резольвенте путем замены текущей переменной на соответствующий терм. В результате применения подстановки некоторые переменные конкретизируются значениями, которые (значения) могут и будут далее использованы при доказательстве истинности тела выбранного правила.

## **2.10 В каких случаях запускается механизм отката?**

Механизм отката запустится в случае неудачи алгоритма унификации.

## **2.11 Когда останавливается работа системы?**

Работа системы останавливается, когда найдены все возможные ответы на вопрос.

## **2.12 Как это определяется на формальном уровне?**

Когда в резольвенте находится исходный вопрос, для которого пройдена вся БЗ.