



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 № 6

по курсу «Функциональное и логическое программирование»

на тему: «Использование функционалов»

Студент ИУ7-61Б
(Группа)

(Подпись, дата)

Мицевич М. Д.
(И. О. Фамилия)

Преподаватель

(Подпись, дата)

Толшинская Н. Б.
(И. О. Фамилия)

2022 г.

Задание 1

Постановка задачи

Написать хвостовую рекурсивную функцию `my-reverse`, которая развернет верхний уровень своего списка-аргумента `lst`.

```
(defun move-to (lst res) (cond ((null lst) result)
  (T (move-to (cdr lst) (cons (car lst) result)))))
```

Задание №2

Написать функцию, которая возвращает первый элемент списка-аргумента, который сам является непустым списком.

```
(defun flistp (lst) (cond ((null (car lst)) Nil)
  ((listp (car lst)) (car lst))
  (T (fnotnull(cdr lst)))))
```

Задание №3

Написать функцию, которая выбирает из заданного списка только те числа, которые больше 1 и меньше 10.

(Вариант: между двумя заданными границами.)

Решение

Листинг 1 – Решение задания №3

```
(defun select-between (lst left right) (cond ((null lst) Nil)
  ((and (< (car lst) right) (> (car lst) left)) (nconc (list (car
    lst)) (select-between (cdr lst) left right)))
  (T (select-between (cdr lst) left right)) ))
```

Задание №4

Напишите рекурсивную функцию, которая умножает на заданное число-аргумент все числа из заданного списка-аргумента.

Решение

Все элементы списка — числа.

Листинг 2 – Решение задания №3

```
(defun prod (lst num) (cond ((null lst) Nil)
  (T (nconc (list (* (car lst) num)) (prod (cdr lst) num))))))
```

Элементы списка – любые объекты.

Листинг 3 – Решение задания №3

```
(defun prod (lst num) (cond ((null lst) Nil)
  ((listp (car lst)) (nconc (list (prod (car lst) num)) (prod (cdr
    lst) num))))
  (T (nconc (list (* (car lst) num)) (prod (cdr lst) num))))))
```

Задание №5

Напишите функцию, `select-between`, которая из списка-аргумента, содержащего только числа, выбирает только те, которые расположены между двумя указанными границами- аргументами и возвращает их в виде списка (упорядоченного по возрастанию списка чисел (+ 2 балла)).

Решение

Листинг 4 – Решение задания №5

```
(defun select-between (lst left right) (cond ((null lst) Nil)
  ((and (< (car lst) right) (> (car lst) left)) (nconc (list (car
    lst)) (select-between (cdr lst) left right)))
  (T (select-between (cdr lst) left right)) ))
```

Задание №6

Написать рекурсивную версию (с именем `rec-add`) вычисления суммы чисел заданного списка.

Решение

Одноуровневый смешанный список.

Листинг 5 – Решение задания №5

```
(defun add (lst res) (cond ((null lst) res)
  ((numberp (car lst)) (add (cdr lst) (+ (car lst) res))))
```

```
(t (add (cdr lst) res))))  
  
(defun rec-add (lst) (add lst 0))
```

Структурированный список.

Листинг 6 – Решение задания №5

```
(defun add (lst res) (cond ((null lst) res)  
  ((numberp (car lst)) (add (cdr lst) (+ (car lst) res)))  
  ((listp (car lst)) (let ((tmp (add (car lst) 0))) (add (cdr lst)  
    (+ tmp res)) ))  
  (t (add (cdr lst) res))))  
  
(defun rec-add (lst) (add lst 0))
```

Задание №7

Написать рекурсивную версию с именем recnth функции nth.

Листинг 7 – Решение задания №5

```
(defun recnth (lst ind) (cond ((null lst) nil)  
  ((= ind 0) (car lst))  
  (T (recnth (cdr lst) (1- ind)))))
```

Задание №8

Написать рекурсивную функцию allodd, которая возвращает t когда все элементы списка нечетные.

Решение

Листинг 8 – Решение задания №5

```
(defun sum-sizes (lst)  
  
  (defun sub (lst res) (cond ((null lst) res)  
    (T (sub (cdr lst) (and (oddp (car lst)) res)))))  
  
  (defun allodd (lst) (sub lst T))
```

Задание №9

Написать рекурсивную функцию, которая возвращает первое нечетное число из списка (структурированного), возможно создавая некоторые вспомогательные функции.

Решение

Листинг 9 – Решение задания №5

```
(defun sum-sizes (lst)
  (defun fodd (lst) (cond ((null lst) nil)
    ((numberp (car lst)) (if (oddp (car lst)) (car lst) (fodd (cdr
      lst)))))
  ((listp (car lst)) (let ((tmp (fodd (car lst)))) (if (null tmp) (
    fodd (cdr lst)) tmp)))
  (T (fodd (cdr lst)))))
```

Задание №10

Используя cons-дополняемую рекурсию с одним тестом завершения, написать функцию которая получает как аргумент список чисел, а возвращает список квадратов этих чисел в том же порядке

Решение

Листинг 10 – Решение задания №5

```
(defun sqr (lst res) (cond ((null lst) res)
  ((numberp (car lst)) (sqr (cdr lst) (cons (* (car lst) (car lst))
    res))))
((listp (car lst)) (sqr (cdr lst) (cons (sqr (reverse (car lst))
  ())) res)))
(T (sqr (cdr lst) res))))

(defun my-sqr (lst) (sqr (reverse lst) ()))
```

Листинг 11 – Решение задания №9

```
; 1
```