



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 № 4

по курсу «Функциональное и логическое программирование»

на тему: «Использование управляющих структур, работа со списками»

Студент ИУ7-66Б
(Группа)

(Подпись, дата)

Жаворонкова А. А.
(И. О. Фамилия)

Преподаватель

(Подпись, дата)

Толпинская Н. Б.
(И. О. Фамилия)

Преподаватель

(Подпись, дата)

Строганов Ю. В.
(И. О. Фамилия)

2024 г.

Практическая часть

Листинг 1 – Исходный код заданий лабораторной работы

```
1 ;; 1. Каковы результаты вычисления следующих выражений?
2 (setf lst1 '(a b c))
3 (setf lst2 '(d e))
4 (cons lst1 lst2)           ;; --> ((a b c) d e)
5 (list lst1 lst2)          ;; --> ((a b c) (d e))
6 (append lst1 lst2)        ;; --> (a b c d e)
7
8 ;; 2. Каковы результаты вычисления следующих выражений?
9 (reverse '(a b c))         ;; --> (c b a)
10 (reverse '(a b (c (d)))) ;; --> ((c (d)) b a)
11 (reverse '(a))            ;; --> (a)
12 (reverse ())              ;; --> Nil
13 (reverse '((a b c)))      ;; --> ((a b c))
14
15 (last '(a b c))           ;; --> (c)
16 (last '(a))               ;; --> (a)
17 (last '((a b c)))         ;; --> ((a b c))
18 (last '(a b (c)))         ;; --> ((c))
19 (last ())                 ;; --> Nil
20
21 ;; 3. Написать, по крайней мере, 2 варианта функции, которая
    возвращает последний элемент своего списка-аргумента
22 (defun f3-1(lst)
23   (car (reverse lst)))
24 (defun f3-2(lst)
25   (cond ((= (length lst) 1)
26         (car lst))
27         (T (f3-2 (cdr lst)))))
28
29 ;; 4. Написать, по крайней мере, 2 варианта функции, которая
    возвращает свой список-аргумент без последнего элемента
30 (defun f4-1(lst)
31   (reverse (cdr (reverse lst))))
32 (defun f4-2(lst)
33   (cond ((> (length lst) 1)
34         (cons (car lst) (f4-2 (cdr lst)))))
35
36 ;; 5. Написать функцию swap-first-last, которая переставляет
```

```

переставляет в списке-аргументе первый и последний элементы
37 (defun f5(lst)
38   (let ((a (first lst))
39         (b (car (last lst))))
40     (reverse (rplaca (reverse (rplaca lst b)) a))))
41
42 ;; 6. Написать простой вариант игры в кости
43 (defun throw-dices-1()
44   (and (setf s1 (+ 1 (random 5))) (setf s2 (+ 1 (random 5)))))
45 (defun throw-dices-2()
46   (and (setf s3 (+ 1 (random 5))) (setf s4 (+ 1 (random 5)))))
47 (defun is-winner(a b)
48   (or (= 7 (+ a b)) (= 11 (+ a b))))
49 (defun is-retry(a b)
50   (or (and (= a 1) (= b 1)) (and (= a 6) (= b 6))))
51 (defun print-throw(a b)
52   (and (print 'throw) (print a) (print b) T))
53
54 (defun f6()
55   (cond ((and (throw-dices-1) (print-throw s1 s2) (is-winner
56     s1 s2))
57     'player_1_won)
58     ((is-retry s1 s2)
59     (f6))
60     ((and (throw-dices-2) (print-throw s3 s4) (is-winner
61     s3 s4))
62     'player_2_won)
63     ((> (+ s1 s2) (+ s3 s4))
64     'player_1_won)
65     ((< (+ s1 s2) (+ s3 s4))
66     'player_2_won)
67     (T 'draw))
68 )
69
70 ;; 7. Написать функцию, которая по своему списку-аргументу
определяет является ли он палиндромом
71 (defun cmp(lst1 lst2)
72   (cond ((and (= 1 (length lst1)) (= 1 (length lst2)))
73     (eq (car lst1) (car lst2)))
74     ((not (eq (car lst1) (car lst2)))
75     Nil)

```

```

74         (T (cmp (cdr lst1) (cdr lst2))))))
75
76 (defun f7(lst)
77   (cmp lst (reverse lst)))
78
79 ;; 8. Написать свои необходимые функции, которые обрабатывают
80    таблицу из 4х точечных пар
81 (defun my-assoc(el tbl)
82   (cond ((= 0 (length tbl))
83          Nil)
84         ((eq (caar tbl) el)
85          (car tbl))
86         (T (my-assoc el (cdr tbl)))))
87
88 (defun my-rassoc(el tbl)
89   (cond ((= 0 (length tbl))
90          Nil)
91         ((eq (cdar tbl) el)
92          (car tbl))
93         (T (my-rassoc el (cdr tbl)))))
94
95 ;; 9. Написать функцию, которая умножает на заданное
96    число-аргумент первый числовой элемент списка из заданного 3х
97    элементного списка-аргумента, когда а) все элементы списка
98    --- числа; б) элементы списка --- любые объекты.
99 (defun f9-1(n lst)
100   (rplaca lst (* n (car lst))))
101 (defun f9-2(n lst)
102   (cond ((= 0 (length lst))
103          Nil)
104         ((numberp (car lst))
105          (rplaca lst (* n (car lst))))
106         (T (f9-2 n (cdr lst)))))

```