



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 №5 по курсу «Функциональное и логическое программирование»

Тема Использование управляющих структур, работа со списками

Студент Климов И.С.

Группа ИУ7-62Б

Оценка (баллы) _____

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

Москва — 2022 г.

Задание 1

Написать функцию, которая по своему списку-аргументу `lst` определяет является ли он палиндромом (то есть равны ли `lst` и `(reverse lst)`).

Решение

```
(defun palindrome (lst)
  (cond ((null (listp lst)) NIL)
        (T (equal lst (reverse lst)))))
```

Задание 2

Написать предикат `set-equal`, который возвращает `t`, если два его множества-аргумента содержат одни и те же элементы, порядок которых не имеет значения.

Решение

```
(defun set_equal (lst1 lst2)
  (and (listp lst1) (listp lst2)
        (subsetp lst1 lst2) (subsetp lst2 lst1)))
```

Задание 3

Напишите свои необходимые функции, которые обрабатывают таблицу из 4-х точечных пар: (страна . столица), и возвращают по стране – столицу, а по столице – страну .

Решение

```
(defun capital_by_country (table country)
  (cond ((null table) NIL)
        ((equal (caar table) country) (cdar table))
        (T (capital_by_country (cdr table) country))))
```

```
(defun country_by_capital (table capital)
  (cond ((null table) NIL)
        ((equal (cdar table) capital) (caar table))
        (T (country_by_capital (cdr table) capital))))
```

Задание 4

Напишите функцию `swap-first-last`, которая переставляет в списке-аргументе первый и последний элементы.

Решение

```
(defun swap_first_last (lst)
  (if (null (listp lst)) NIL)
  (let ((first_elem (car lst)))
    (setf (car lst) (car (last lst)))
    (setf (car (last lst)) first_elem)
    lst))
```

Задание 5

Напишите функцию `swap-two-elements`, которая переставляет в списке-аргументе два указанных своими порядковыми номерами элемента в этом списке.

Решение

```
(defun swap_two_elements (lst i j)
  (if (null (listp lst)) NIL)
  (let ((i_elem (nth i lst)))
    (setf (nth i lst) (nth j lst))
    (setf (nth j lst) i_elem)
    lst))
```

Задание 6

Напишите две функции, `swap-to-left` и `swap-to-right`, которые производят одну круговую перестановку в списке-аргументе влево и вправо, соответственно.

Решение

```
(defun swap_to_left (lst)
  (cond ((null (listp lst)) NIL)
        ((< (length lst) 2) lst)
        (T (let ((first_elem (first lst)))
              (setf (car lst) (second lst))
              (setf (cdr lst) (append (nthcdr 2 lst) (list first_elem)))
              lst))))

(defun swap_to_right (lst)
  (cond ((null (listp lst)) NIL)
        ((< (length lst) 2) lst)
        (T (let ((last_elem (car (last lst))))
              (setf (cdr lst) (butlast lst))
              (setf (car lst) last_elem)
              lst))))
```

Задание 7

Напишите функцию, которая добавляет к множеству двухэлементных списков новый двухэлементный список, если его там нет.

Решение

```
(defun add_two_element_list (lst new_pair)
  (cond ((null (listp lst)) NIL)
        ((null (notany #'(lambda (pair) (equal pair new_pair)) lst)) NIL)
        (T (nconc lst (list new_pair)))))
```

Задание 8

Напишите функцию, которая умножает на заданное число-аргумент первый числовой элемент списка из заданного 3-х элементного списка-аргумента, когда

- a. все элементы списка – числа,
- b. элементы списка – любые объекты.

Решение

```
; all elements are numbers
(defun mul_first_number_v1 (lst k)
  (cond ((or (null (listp lst)) (null (numberp k))) NIL)
        (T (setf (car lst) (* (car lst) k))))
  lst)

; all elements are objects
(defun mul_first_number_v2 (lst k)
  (cond ((or (null (listp lst)) (null (numberp k))) NIL)
        (T (let ((stop 0)
                  (i 0))
              (mapcar #'(lambda (element) (if (and (numberp element) (= stop 0))
                                                (setf stop 1
              (nth i lst) (* element k))
              (setf i (+ i 1)))) lst))
  lst)))
```

Задание 9

Напишите функцию, `select-between`, которая из списка-аргумента из 5 чисел выбирает только те, которые расположены между двумя указанными границами-аргументами и возвращает их в виде списка (упорядоченного по возрастанию списка чисел (+ 2 балла)).

Решение

```
(defun select_between (lst left right)
  (cond ((or (null (listp lst))
             (null (numberp left)) (null (numberp right))) NIL)
        (T (if (> left right) (let ((tmp left))
                                (setf left right)
                                (setf right tmp)))
            (sort (remove NIL (mapcar #'(lambda (num)
                                           (if (and (> num left) (< num right)) num))
                    lst))
                  #'<))))
```