

|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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»\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Лабораторная работа № 8**

|  |  |
| --- | --- |
| **Тема**  Lisp  **Студент** Белоусова Ю.С.  **Группа** ИУ7-61Б  **Оценка (баллы) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Преподаватель** Толпинская Н.Б. |  |

Москва.

2020 г.

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

(defun palindrome (lst) (equal lst (reverse lst)) )

Встроенная функция reverse оборачивает список, не разрушая его структуру, т.е. копирует весь список и все изменения происходят над копией списка. Аналог разрушающей функции - nreverse. В данном случае его использовать нельзя, поскольку в результате вычисления (nreverse lst) изменится и сам список lst, сравнение будет происходить между одним и тем же объектом и результат вычисления функции всегда будет равен T.

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

(defun swap-first-last (lst)(nconc (nconc (last lst) (cdr (nbutlast lst))) (cons (car lst) nil)))

Встроенная функция nbutlast вычисляет список без последнего его элемента, разрушая исходный список. Объединяя в список последний элемент исходного списка, результат работы этой функции (без головы) и голову исходного списка с cdr указателем на Nil, получаем необходимый результат.

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

(defun swap-two-element (lst first second) (let ((temp (nth first lst))) (setf (nth first lst) (nth second lst )) (setf (nth second lst) temp)) lst)

Используя возможность определения локальной переменной, объявляем временный буфер temp, в который записывается значение элемента списка с индексом first. Далее элементу списка с индексом first присваивается значение элемента с индексом second, а элементу с индексом first значение из временного буфера temp.

1. Напишите две функции, swap-to-left и swap-to-right, которые производят круговую перестановку в списке-аргументе влево и вправо, соответственно (на k позиций).
2. (defun swap-to-left (lst k) (nconc (subseq lst k) (nconc (subseq lst 0 k) nil)))
3. (defun swap-to-right (lst k) (let ((ind (- (length lst) k))) (nconc (subseq lst ind) (nconc (subseq lst 0 ind) nil))))

Встроенная функция (subseq lst start end) создает новый список на основе исходного lst, начиная с индекса start и по end (не включительно). Если указан только один аргумент помимо списка, то функция создает список из элементов, начиная с индекса = аргументу и до конца списка. Таким образом, чтобы передвинуть на k позиций список влево или вправо нужно соединить в один список результаты работы subseq для k и для (length lst) - k соответственно. Для соединения в один список использовалась разрушающая структуру функция nconc.

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).

(defun select-between (lst left right) (sort (remove-if #'(lambda (x) (or (< x left) (> x right))) lst) #'<))

Функционал remove-if использует предикат, переданный как функция, чтобы создать список на основе исходного, не включая в него элементы, для которых предикат вычислил значения T. Для решения задачи из исходного списка формируется новый, не содержащий элементы, не входящие в указанный интервал. Далее полученный список сортируется по возрастанию с помощью функционала sort.