

Федеральное государственное бюджетное образовательное учреждение высшего образования
«Сибирский государственный университет телекоммуникаций и информатики»
(СибГУТИ)

Кафедра прикладной математики и кибернетики

Лабораторная работа № 2
по дисциплине «Функциональное и логическое программирование»

Бригада №2

Выполнили:
студент группы ИБ-823
Лозовой Владислав
Александрович
ФИО студента

студент группы ИБ-823
Шиндель Эдуард
Дмитриевич
ФИО студента

Работу проверил:
ассистент кафедры Агалаков А.А.
ФИО преподавателя

Новосибирск 2020 г.

ОГЛАВЛЕНИЕ

ОГЛАВЛЕНИЕ	2
ЗАДАНИЕ	3
ДЕМОНСТРАЦИЯ РАБОТЫ ПРОГРАММЫ	3
ВЫВОД	3
ПРИЛОЖЕНИЕ	5

ЗАДАНИЕ

Определить функцию:

1. Преобразующую список в "двойной" список таким образом, чтобы каждый элемент удвоился.

Например, (1 a b 3) → (1 1 a a b b 3 3)

2. Возвращающую список из пар соседних элементов, одинаково отстоящих от начала и конца списка.

Например, (1 a b c 2 3 f) → ((1 f) (a 3) (b 2) (c))

3. Удаляющую элементы из первого списка с номерами из второго списка (второй список упорядочен по возрастанию, нумерация элементов должна начинаться с 1).

Например, L1=(a s d f g h j k l), L2=(1 4 5 8) → (s d h j l)

ДЕМОНСТРАЦИЯ РАБОТЫ ПРОГРАММЫ

```
1 ;task1
2 (defun func (x) (cond ((null x) nil) (t (cons (car x) (cons (car x) (func (cdr x)))))))
3 (func `(1 a b 3))
4
5 ;task2
6 (defun func (x) (cond((null (cdr x)) (list x)) (t (append (list(cons (car x) (last x))) (func (butlast (cdr x))))))
7 (func `(1 a b c 2 3 f))
8
9 ;task3
10 (defun func (x y &optional (n 1)) (cond ((null x) nil) ((null y) x) ((= (car y) n) (func (cdr x) (cdr y) (1+ n))) ((cons (car x) (func (cdr x) (cdr y) (1+ n)))))
11 (func `(a s d f g h j k l) `(1 4 5 8))

[1]> (defun func (x) (cond ((null x) nil) (t (cons (car x) (cons (car x) (func (cdr x)))))))
FUNC
[2]> (func `(1 a b 3))
(1 1 A A B B 3 3)
_ _ _ _ _
(funcun func (x) (cond((null (cdr x)) (list x)) (t (append (list(cons (car x) (last x))) (func (butlast (cdr x)))))))
FUNC
Break 2 [8]> (func `(1 a b c 2 3 f))
((1 F) (A 3) (B 2) (C))
Break 2 [8]> |
(funcun func (x y &optional (n 1)) (cond ((null x) nil) ((null y) x) ((= (car y) n) (func (cdr x) (cdr y) (1+ n))) ((cons (car x) (func (cdr x) (cdr y) (1+ n)))))
FUNC
Break 3 [9]> (func `(a s d f g h j k l) `(1 4 5 8))
(S D H J L)
Break 3 [9]>
```

ВЫВОД

Было выполнено 3 задачи в LISP по написанию функций без использования set, let иsetq. \

Написаны функции:

1. Для преобразования списка в двойной список
2. Возвращающая список из пар соседних элементов, отстоящих от начала и конца списка
3. Удаляющая элементы из первого списка с номерами из второго

ПРИЛОЖЕНИЕ

```
;task1
(defun func (x) (cond ((null x) nil) (t (cons (car x) (cons
(car x) (func (cdr x)))))))
(func `(1 a b 3))

;task2
(defun func (x) (cond((null (cdr x)) (list x)) (t (append
(list(cons (car x) (last x))) (func (butlast (cdr x)))))))
(func `(1 a b c 2 3 f))

;task3
(defun func (x y &optional (n 1)) (cond ((null x) nil) ((null
y) x) ((= (car y) n) (func (cdr x) (cdr y) (1+ n))) ((cons (car
x) (func (cdr x) y (1+ n)))))
(func `( a s d f g h j k l) `(1 4 5 8))
```