Федеральное государственное бюджетное образовательное учреждение высшего образования

«Сибирский государственный университет телекоммуникаций и информатики»

(СибГУТИ)

Кафедра прикладной математики и кибернетики

Лабораторная работа № 3

по дисциплине «Функциональное и логическое программирование»

Бригада №2

Выполнили:

студент группы ИВ-823

Лозовой Владислав Александрович

ФИО студента

студент группы ИВ-823

Шиндель Эдуард Дмитриевич

ФИО студента

Работу проверил:

ассистент кафедры Агалаков А.А.

ФИО преподавателя

Новосибирск 2020 г.

# ОГЛАВЛЕНИЕ

[ОГЛАВЛЕНИЕ 2](#_Toc52186895)

[ЗАДАНИЕ 3](#_Toc52186896)

[ДЕМОНСТРАЦИЯ РАБОТЫ ПРОГРАММЫ 3](#_Toc52186897)

[ВЫВОД 3](#_Toc52186898)

[ПРИЛОЖЕНИЕ 5](#_Toc52186899)

# 

# 

# ЗАДАНИЕ

1. **Определите предикат, проверяющий:**

Является ли одно множество подмножеством другого.

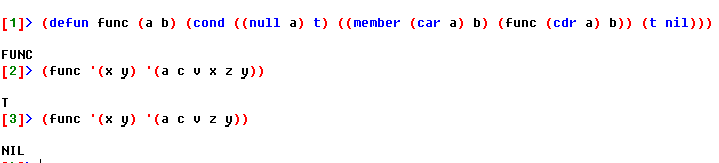
1. **Определите функцию:**

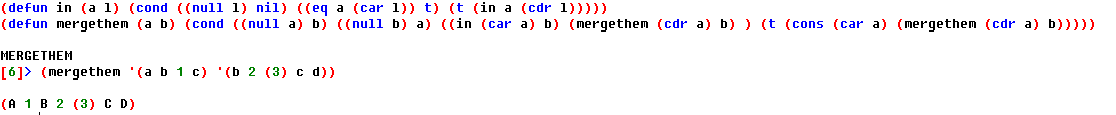
Возвращающую объединение двух множеств.

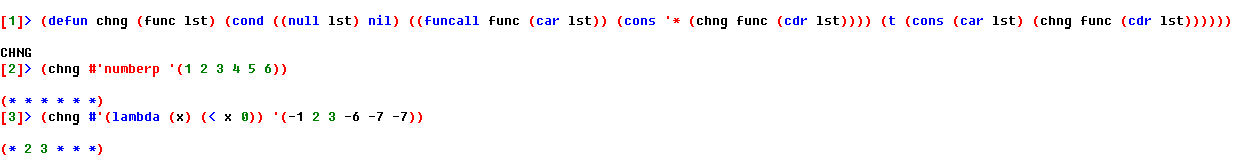
1. **Определите функционал:**

Заменяющий все элементы списка, не обладающие определенным свойством, на символ \*. Проверьте работу функционала для предикатов: - число (функциональный аргумент – имя встроенного предиката numberp); - отрицательное число (функциональный аргумент – лямбда выражение).

**ДЕМОНСТРАЦИЯ РАБОТЫ ПРОГРАММЫ**







**ВЫВОД**

Было выполнено 3 задачи в LISP по написанию функций без использования set, let и setq.

Написаны функции:

1. Для определения, является ли одно множество подмножеством другого
2. Возвращающую объединение двух множеств
3. Заменяющую все элементы списка, не обладающие определенным свойством, на символ \*

# ПРИЛОЖЕНИЕ

;task 1

(defun func (a b) (cond ((null a) t) ((member (car a) b) (func (cdr a) b)) (t nil)))

(func '(x y) '(a c v x z y))

(func '(x y) '(a c v z y))

;task 2

(defun in (a l) (cond ((null l) nil) ((eq a (car l)) t) (t (in a (cdr l)))))

(defun mergethem (a b) (cond ((null a) b) ((null b) a) ((in (car a) b) (mergethem (cdr a) b) ) (t (cons (car a) (mergethem (cdr a) b)))))

(mergethem '(a b 1 c) '(b 2 (3) c d))

;task 3

(defun chng (func lst) (cond ((null lst) nil) ((funcall func (car lst)) (cons '\* (chng func (cdr lst)))) (t (cons (car lst) (chng func (cdr lst))))))

(chng #'numberp '(1 2 3 4 5 6))

(chng #'(lambda (x) (< x 0)) '(-1 2 3 -6 -7 -7))