|  |  |
| --- | --- |
| 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 **ПРОГРАМНОЕ ОБЕСПЕЧЕНИЕ ЭВМ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ (ИУ7)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.04.03 ПРОГРАММНАЯ ИНЖЕНЕРИЯ**

**Отчет**

|  |  |
| --- | --- |
| **По лабораторной работе №** | 9 |

**Дисциплина:** Функциональное и логическое программирование

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ7-62Б |  |  | Н.А. Гарасев |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | Н.Б.Толпинская |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2021

**Задание 4. Выбрать только числа больше 1 и меньше 10.**

**(defun** find-num **(**lst**)**

**(**remove-if-not **(lambda** **(**a**)** **(and** **(<** a 10**)** **(>** a 1**)))** lst**))**

\* (find-num `(1 11 21 31 4 45 5 65))

(4 5)

**Задание 5. Декартово произведение двух.**

; Декартово произведение

**(defun** decart **(**lstX lstY**)**

**(mapcan** #**'(lambda** **(**x**)**

**(mapcar** #**'(lambda** **(**y**)**

**(list** x y**))** lstY**))** lstX**))**

\* (decart `(1 2 3 4) `(5 6 7 8))

((1 5) (1 6) (1 7) (1 8) (2 5) (2 6) (2 7) (2 8) (3 5) (3 6) (3 7) (3 8) (4 5)

(4 6) (4 7) (4 8))

**Задание 6.** Reduce первоначально инициализирует буфер нулем, поэтому при вызове (reduce #`+ ()) результатом будет 0. Можно изменить это значение параметром initial-value, например: (reduce #'+ () :initial-value nil) -> NIL.

**Задание 7. Сумма длин списка списков.**

**(defun** my\_len **(**lst**)**

**(cond** **((null** lst**)** 0**)**

**(**t **(+** 1 **(**my\_len**(cdr** lst**))))))**

**(defun** len-list-inner **(**lst len**)**

**(cond** **((null** lst**)** len**)**

**(**T **(**len-list-inner **(cdr** lst**)** **(+** **(**my\_len **(car** lst**))** len**)))))**

**(defun** len-list **(**lst**)**

**(cond** **((null** lst**)** 0**)**

**(**T **(**len-list-inner lst 0**))))**

\* (len-list `((1 2 3 4 5) (1 2 3) (1 2) (1)))

11

**Дополнительное задание. Список смешанного структурирования, квадраты.**

**(defun** squadr **(**lst**)**

**(mapcar** **(lambda** **(**a**)**

**(cond** **((listp** a**)** **(**squadr a**))**

**((numberp** a**)** **(\*** a a**))**

**(**T a**)))** lst**))**

\* (squadr `(1 2 (3 (4 (5)) 6) (7 (8 9))))

(1 4 (9 (16 (25)) 36) (49 (64 81)))

**Вопросы.**

**Вопрос 1. Классификация рекурсивных функций.**

Классифицировать рекурсии можно несколькими способами.

Первый способ классификации:

* Простая рекурсия
* Рекурсия первого порядка
* Взаимная рекурсия

Другая классификация:

* Хвостовая рекурсия
* Дополнительная рекурсия