Описание и вызов функций в языке Лисп.

Описание простейших рекурсивных функций в языке Лисп.

**Результаты выполнения заданий: (Вариант 3)**

**Задание 1**.

Описать неименованную функцию для объединения голов трех списков в один список, исходные данные взять из таблицы 4.

(print ((lambda (X Y Z)

(list (car X) (car Y) (car Z)))

'('(PI) V '(H J K)) '(R YU '(H KJ KL)) '(U II OO LL PP '(3 4 5))

))

Результат:

('(PI) R U)

[Finished in 0.043s]

Обоснование:

Функция list() использовалась для объединения параметров в список, а функция car() для получения первого элемента списка.

**Задание 2.**

Описать именованную функцию для создания нового списка из элементов нескольких исходных списков. В качестве исходных списков использовать списки таблицы 4. Номера элементов списков взять в таблице 5.

Результат:

('(PI) '(H KJ KL) '(3 4 5))

[Finished in 0.052s]

Обоснование:

Помимо использования стандартных функций list() и car(), также использовалась nth() для выбора конкретного элемента из списка.

**Задание 3.**

Описать именованную функцию в соответствии с вариантом индивидуального задания в Таблице 6:

Есть пять чисел. Написать функцию , формирующую список из максимального и минимального по модулю чисел, если минимальное и максимальное числа – целые, среднее арифметическое минимального и максимального чисел – иначе.

Результат:

(8 1)

4.7

[Finished in 0.082s]

Обоснование:

Для проверки числа на целый тип было решено вычислить остаток от деления с помощью функции mod(). Для того, чтобы взять каждое число по модулю был использовалась функция abs().

**Задание 4.**

Описать функцию в соответствии со своим вариантом задания из Таблицы 1:

Описать функцию, которая находила бы сумму всех числовых элементов списка с учетом наличия подсписков. Пример : для списка ‘(1 ((2 3) 4) 5 6) результатом будет 21.

Результат:

21

[Finished in 0.047s]

Обоснование:

Стандартные функции car() и cdr() использовались для использования головы и хвоста списка.