Министерство науки и высшего образования РФ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Новгородский государственный университет имени Ярослава Мудрого»

Кафедра информационных технологий и систем

**Локальные определения**

Лабораторная работа по дисциплине:

«Функциональное и логическое программирование»

Выполнил студент группы 8091:

\_\_\_\_\_\_\_\_\_\_ / Васильев И.В.

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

Принял преподаватель:

\_\_\_\_\_\_ / Михайлов Д.В.

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

**Великий Новгород**

**2021**

1. **Цель и задачи.**

Целью работы является практическое изучение различных видов локальных определений и особенностей их использования в рекурсивных программах.

Основные задачи:

− изучить применение техники нисходящей и восходящей рекурсии при написании рекурсивных функций с использованием локальных определений;

− сравнить возможности локальных определений LET и LAMBDA по организации вычислений в рекурсивных программах;

Задание №1

Описать функцию вычисления факториала. Рассмотреть варианты решения задачи с применением локальных определений LAMBDA и LET.

Задание №2

Разработать программу символьного дифференцирования в соответствии с правилами, изложенными в [8]. Рассмотреть варианты решения задачи с применением локальных определений LAMBDA и LET.

Задание №3

Решить задачу из лабораторной работы №2 с применением локальных определений LAMBDA и LET.

Задание №4  
Реализовать простейший интерпретатор лисповских программ. На вход интерпретатора подается текст, который может быть интерпретирован как вызов или суперпозиция функций Лиспа, пример (для Common Lisp): ‘(cons (car (cdr ‘(e r t w))) (cons (cdr ‘(g h 6)) ‘())). Программа должна обеспечивать выполнение такого рода примеров. Требования к программе: − интерпретация базовых функций Лиспа и арифметических операций +, -, /, \*; − в программе должны использоваться локальные определения; − не допускается использование встроенной функции-интерпретатора EVAL;

Задание №5

Дополнить интерпретатор из задания 4 в соответствии с вариантом индивидуального задания из Таблицы 1.

1. **Решение**

Задание №1

Код решения задачи приложен в архиве в файле “Васильев Lab41”

Вывод программы:



Рисунок 1 при (fac1 5) (fac2 5)

Задание №2

Теория:

Символьным дифференцированием в математике называется операция преобразования одного арифметического выражения в другое арифметическое выражение, которое называется производной.

Так как в языке Lisp используется префиксная запись, я использовал следующую таблицу:

Таблица преобразований:

|  |  |
| --- | --- |
| C | 0 |
| x | 1 |
| ^X C | (+ С (^ X (- C 1))) |
| (+ U V) | (+ U’ V’) |
| (+ U V) | (- U’ V’) |
| (\* U V) | (+ (\*U’ V) (\* U V’)) |
| (/ U V) | (/(- (\*U’ V) (\* U V’)) V^2) |

Код решения задачи приложен в архиве в файле “Васильев Lab42”

Вывод программы:

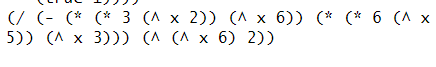


Рисунок 2 При: (differentiation '(/ (^ x 3) (^ x 6)) 'x)

Задание №3

Описать функцию, которая на основе списка чисел формирует список-результат следующим образом : первый элемент есть произведение элементов списка, второй – произведение элементов хвоста, третий – произведение элементов хвоста хвоста и так далее.  
В решении задачи была применена нисходящая рекурсия, для вычисления промежуточных результатов (произведение последующих, без первого элемента), что позволило разбить задачу на несколько подзадач, возвращая значения.

Код решения задачи приложен в архиве в файле “Васильев Lab43”

Вывод программы:



Рисунок 3 При (lab4let '(1 2 3 4 5 6)) (lab4lambda '(1 2 3 4 5 6))

Задание №4-5

Теория:

Реализация интерпретатора, на вход подаётся функция или суперпозиция функций, программа анализирует входные данные и подсчитывает результат.

Реализованные функции:

* +
* -
* /
* \*
* First
* Rest
* Cons
* Acos

Программа работает по следующему принципу: сравнивает входную функцию с заданными и переходит к её анализу, если встречается не атом, переход на анализ функции. После выполнения подфункций, возвращается результат, тем самым получается возможность считать значение как атом.

Код решения задачи приложен в архиве в файле “Васильев Lab44”

Вывод программы:

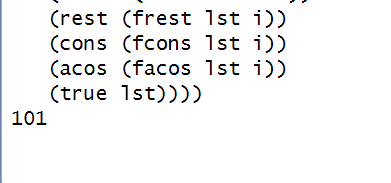


Рисунок 4 (interpritator '(+ (- (\* (/ 10 2)2)9)100))

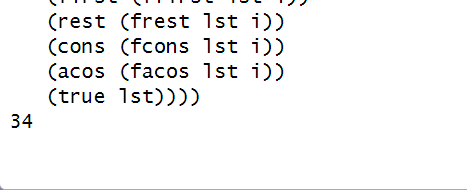


Рисунок 5 (interpritator '(first (rest (10 (+ 33 1) (- 5 1) 5 (+ 10 1)))55))

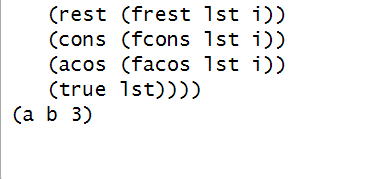


Рисунок 6 (interpritator '(cons a b (+ 1 2)))

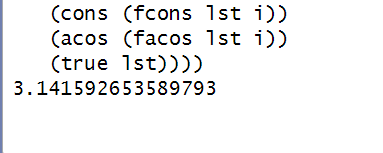


Рисунок (interpritator '(acos (- 0 1)))

1. **Вывод:**

В ходе выполнения лабораторной работы я научился применять техники нисходящей и восходящей рекурсии при написании рекурсивных функций с использованием локальных переменных.