Отчет по лабораторной работе № 1 по курсу «Функциональное программирование»

Студент группы М8О-307 МАИ *Днепров Иван*, №10 по списку
Контакты: vanya.dneprov@gmail.com
Работа выполнена: 15.03.2020

Преподаватель: Иванов Дмитрий Анатольевич, доц. каф. 806
Отчет сдан:
Итоговая оценка:
Подпись преподавателя:

1. Тема работы

Примитивные функции и особые операторы Common Lisp.

2. Цель работы

Научиться вводить S-выражения в Лисп-систему, определять переменные и функции, работать с условными операторами, работать с числами, использую схему линейной и древовидной рекурсии.

3. Задание (вариант №1.44)

Запрограммируйте на языке Коммон Лисп функцию, вычисляющую элементы треугольника Паскаля. Единственный параметр - натуральное число, задающее "глубину"треугольника.

Реализуйте рекурсивно-итеративный процесс. Для выдачи на печать используйте функции print и prin1.

4. Оборудование студента

MacBook (13-inch, Mid 2010), процессор 2,4 GHz Intel Core 2 Duo, память: 8Gb, разрядность системы: 64.

5. Программное обеспечение

Mac OS 10.13.6, компилятор clisp, текстовый редактор Sublime Text 3.

6. Идея, метод, алгоритм

Для простоты я решил разбить задачу на две подзадачи. Основная задача – создание новой строки треугольника паскаля по предыдущей (функция newrow). И дополнительная – запуск newrow нужное число раз и вывод полученных результатов (функции print-rows и pascal-triangle).

Идея newrow заключается в том, что в новой строке в начале мы записываем 1, а затем суммы пар элементов предыдущей строки (первый + второй, втрой + третий и т. д.) и её последний элемент.

Функция print-rows вызывет newrow нужное число раз и выводит списки, а pascal-triangle вызывает print-rows с начальным списком.

7. Сценарий выполнения работы

8. Распечатка программы и её результаты

8.1. Исходный код

```
(defun pascal-triangle (n)
   (print-rows n '(1)))

(defun print-rows (n l)
   (when (< 0 n)
       (print l)
       (print-rows (1- n) (cons 1 (newrow l)))))

(defun newrow (l)
   (if (> 2 (length l))
      '(1)
      (cons (+ (car l) (cadr l)) (newrow (cdr l)))))
```

8.2. Результаты работы

```
(pascal-triangle 5)

(1)
(1 1)
(1 2 1)
(1 3 3 1)
(1 4 6 4 1)
NIL

(pascal-triangle -12)
```

```
NIL

(pascal-triangle 0)
NIL

(pascal-triangle 12)

(1)
(1 1)
(1 2 1)
(1 3 3 1)
(1 4 6 4 1)
(1 5 10 10 5 1)
(1 6 15 20 15 6 1)
(1 7 21 35 35 21 7 1)
(1 8 28 56 70 56 28 8 1)
(1 9 36 84 126 126 84 36 9 1)
(1 10 45 120 210 252 210 120 45 10 1)
(1 11 55 165 330 462 462 330 165 55 11 1)
NIL
```

9. Дневник отладки

Дата	Событие	Действие по исправлению	Примечание

10. Замечания автора по существу работы

Хоть это и элиментарное задание, мне пришлось немного подумать над агоритмом, в итоге я создал три функции: newrow – функция, создающая новый список (строку в треугольнике паскаля), print-rows – функция, которая, начиная со списка (1), выводит список и вызывает newrow для создания нового списка (следующей строки треугольника) и pascal-triangle – функция с одним параметром, котрая просто вызывает print-rows с двумя параметрами: заданным в pascal-triangle и списком (1).

11. Выводы

После курса по prolog, который у меня был в третьем семестре, я писал используя только процедурную и объектно-ориентированную парадигмы. Выполнение этой лабораторной работы заставило меня опять посмотреть на программирование под другим углом. И хоть само задание довольно простое, мне пришлось как следует подумать, чтобы его выполнить.