

ЛАБОРАТОРНАЯ РАБОТА 3

МЕТОДЫ РАЗРАБОТКИ ФУНКЦИОНАЛЬНЫХ ПРОГРАММ

1. Цель и задачи.

Целью работы является изучение основных методов разработки функциональных программ с позиций Строго Функционального Языка.

Основные задачи :

- Освоить приемы нисходящего и восходящего проектирования функциональных программ;
- Научиться выделять основные и вспомогательные функции с учетом разбиения задачи на подзадачи;
- Овладеть приемами использования накапливающих параметров во вспомогательных функциях;
- Ознакомиться с упреждающим использованием результата вызова функции.

2. Задание на лабораторную работу.

ВНИМАНИЕ! в лабораторной работе можно использовать только средства строго функционального языка программирования. Иными словами при описании функций нельзя использовать функции SET, SETQ, SETF и циклы. Все функции программ должны быть разработаны самостоятельно.

2.1 Задание 1

Написать программу сортировки списка методом Шелла. Вычисление последовательности шагов сортировки производится в соответствии с вариантом в Таблице 1.

Таблица 1. Вычисление шага сортировки Шелла.

Вариант.	Вычисление последовательности шагов.
1 – 8.	Методом Р. Седжвика, рассмотренным в [6].
9 – 16.	Методом, предложенным Дональдом Кнутом : $h_{k-1} = 3 * h_k + 1$, $h_t = 1$, где h_i - шаг сортировки, $t = \lceil \log_3 n \rceil - 1$ - число шагов сортировки, n - длина списка.
17 - 25.	Методом, предложенным Дональдом Кнутом : $h_{k-1} = 2 * h_k + 1$, $h_t = 1$, где h_i - шаг сортировки, $t = \lceil \log_2 n \rceil - 1$ - число шагов сортировки, n - длина списка.

2.2 Задание 2

Написать программу сортировки [6] списка в соответствии с вариантом в таблице 2.

Таблица 2. Методы сортировки списков.

Вариант.	Реализуемый метод сортировки.
1.	Сортировка простыми включениями.
2.	Сортировка бинарными включениями.
3.	Сортировка методом прямого выбора.
4.	Сортировка методом пузырька.
5.	Шейкер-сортировка.
6.	Сортировка Хоара.

Сравнить эффективность реализованной сортировки и реализованного в Задании 1 варианта сортировки Шелла.

2.3 Задание 3

Написать программу объединения двух отсортированных списков в один. При этом порядок сортировки в списке-результате должен сохраняться.

2.4 Задание 4

Написать программу в соответствии с заданием из Таблицы 3.

Таблица 3. Вариант индивидуального задания.

Вариант.	Задание.
1, 13, 24	Написать программу, возвращающую T, если lst2 является подсписком lst1 глубины N. Элементами списка могут быть атомы и (или) списки любой глубины вложения.
2, 14	Написать функцию, вычисляющую сумму элементов-чисел на каждом уровне исходного списка. Рекомендуется следующая форма результата : ((1 <сумма числовых элементов на первом уровне>)(2 <на втором>)..) Пример : для списка (a (b (4 (2 e (3) k 15) e 5) 7)) результатом будет список : ((1 0)(2 7)(3 9)(4 17) (5 3)).
3, 15	Написать программу, возвращающую список, содержащий информацию о количестве подсписков на каждом уровне вложенности : ((<уровень><количество подсписков>)...).
4, 16	Написать программу, которая в исходном списке заменяет все элементы-символы соответствующими им ASCII-кодами. Список может содержать подсписки произвольной глубины вложения.
5, 17	Написать программу, которая в исходном списке заменяет все элементы-целые числа остатками от их деления на 2. Список может содержать подсписки произвольной глубины вложения.
6, 18	Написать функцию, генерирующую все циклические перестановки списка. Элементами списка являются списки. Пример : ((a b)(c d)) дает (((a b)(c d))((b a)(c d))((a b)(d c))..).
7, 19	Функция должна возвращать список позиций вхождения и глубин нахождения списка lst2 в список lst1.
8, 20	Есть список, написать программу, возвращающую максимальную глубину списка.
9, 21	Написать функцию, удаляющую из исходного списка подсписки заданной глубины.
10, 22	Заданы глубина подсписка, позиция и s-выражение. Включить s-выражение во все имеющиеся подсписки заданной глубины и на заданную позицию.
11, 23	Заданы глубина подсписка и позиция. Удалить из всех имеющихся подсписков заданной глубины элементы, находящиеся на указанной позиции.
12, 24	Написать программу сортировки списка методом Хоара [6].

3. Содержание отчета по лабораторной работе.

Отчет по лабораторной работе должен содержать :

- формулировку цели и задач;
- описание процесса разработки программ. Для каждого задания в обязательном порядке приводится : обоснование выбранных структур функций, условия окончания рекурсии в каждом случае и формирование новых значений аргументов при рекурсивном вызове;
- выводы по проделанной реализации.

Литература.

1. Хювенен Э., Сеппянен Й. Мир Лиспа. В 2-х т. Пер. с финск. – М.: Мир, 1990.
2. Lutz Mueller newLISP For BSDs, Linux, Mac OS X, Solaris and Win32. Users Manual and Reference // http://www.newlisp.org/downloads/manual_frame.html
3. Вирт Н. Алгоритмы + структуры данных = программы : Пер. с англ. – М.: Мир, 1985. С. 74-108
4. Вирт Н. Алгоритмы и структуры данных : Пер. с англ. – СПб.: Невский диалект, 2001. С. 85-163
5. Ахо А.В., Хопкрофт Д.Э., Ульман Д.Д. Структуры данных и алгоритмы : Пер. с англ.: Уч.пос. – М.: Издательский дом “Вильямс”, 2000.
6. Алгоритмы, методы, исходники : сортировка. // <http://algotlist.manual.ru>