

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования

**«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ»
(ТУСУР)**

Кафедра автоматизации обработки информации

УТВЕРЖДАЮ

Зав. кафедрой АОИ

_____ Ю.П. Ехлаков

« ____ » _____ 2016 г.

ФУНКЦИОНАЛЬНОЕ ПРОГРАММИРОВАНИЕ

**Методические указания к лабораторным работам,
самостоятельной работе и курсовой работе**

для студентов направления 09.03.04

«Программная инженерия»

Разработчик:

к.т.н., доцент каф. АОИ

_____ Н.Ю. Салмина

2016

СОДЕРЖАНИЕ

Введение.....	3
1. Лабораторная работа 1. Основы языка Лисп	4
2. Лабораторная работа 2 Работа с функциями.....	8
3. Лабораторная работа 3 Рекурсивные функции	13
4. Лабораторная работа 4. Разработка функциональных программ.....	18
5. Лабораторная работа 5. Функционалы.....	21
6. Лабораторная работа 6. Использование блочных функций.....	24
7. Графы и деревья.....	24
8. Курсовая работа	26
9. Методические указания для выполнения самостоятельной работы	29
10. Список литературы	31

ВВЕДЕНИЕ

Цель дисциплины – ознакомление студентов с основами функционального программирования на примере языка Лисп, принципами структурирования знаний в виде фреймов и семантических сетей, а также формирование у студентов профессиональных знаний и практических навыков по разработке и созданию интеллектуальных систем с помощью языка функционального программирования Лисп.

Связь дисциплины с другими учебными дисциплинами. Предшествующими дисциплинами, формирующими начальные знания, являются: "Дискретная математика", "Информатика и программирование", "Технологии программирования".

Изучение дисциплины рассчитано на два семестра и включает в себя: теоретический раздел (изучение теоретического материала); практический раздел (выполнение лабораторных и контрольных работ); выполнение курсовой работы; итоговый контроль результата изучения дисциплины. Данное пособие содержит в себе методические указания и варианты заданий для лабораторных работ, самостоятельной работы и курсовой работы. В качестве итогового контроля изучения дисциплины является экзамен и дифференцированный зачет по курсовой работе.

Для допуска к сдаче экзамена необходимо выполнить все лабораторные работы. Студент может получить экзамен автоматом в случае, если он набирает необходимое количество баллов по рейтингу. В случае, если студент не набирает необходимое количество баллов по рейтингу, экзамен сдается при наличии допуска.

Подробное положение о рейтинге находится в рабочей программе дисциплины, а также расположено на кафедральном учебном сервере.

1. ЛАБОРАТОРНАЯ РАБОТА 1. ОСНОВЫ ЯЗЫКА ЛИСП.

Цель работы.

Целью данной работы является знакомство со средой работы в XLISP и получение первичных навыков работы в этой среде путем написания простейших выражений с использованием базовых функций Лисп.

Методические указания.

Необходимо ознакомиться со средой работы в LispWorks.

Все задания можно выполнять в диалоговом режиме.

При выполнении первого задания используйте суперпозиции функций CAR и CDR. Помните, что функция CAR определяет голову списка, а функция CDR выделяет хвост списка.

Внимательно следите за балансом скобок!

Для выполнения второго задания используйте арифметические операции и суперпозиции CAR и CDR.

При выполнении третьего задания используйте функцию проверки условий COND, а также, по необходимости другие основные функции Лисп.

Задание на практику.

- 1) записать выражение, выбирающее выделенный элемент списка.
- 2) Записать выражение, выполняющее заданную арифметическую операцию над всеми элементами заданного числового списка (на всех уровнях).
- 3) записать условие, которое проверяет список и выдает истину, если выполняется хотя бы одно из заданных условий. Если ни одно из заданных условий не выполняется, выдать NIL.

Варианты заданий.

Задание 1.

№ варианта	Список
1	(1 2 (3 4) 5 6 7)
2	(((q w e) g f) n)
3	(a (x c) (s v) n q)
4	((1 2 3) (4 5 6))
5	(9 8 (7 6 5) e w)
6	((d) f(h I k) z)
7	(3 (4 5) (j l))
8	(h z ((q) d k l) a)
9	((1) (2) (3) (4) (5))
10	(h ((a s d) l k)
11	(0 (2 g a) 1 5)
12	((b) v (1 2 3) z a)
13	(e ((1) (2)) a s)
14	(r f b c n z)
15	((1 2 3) s (4 5) f)
16	((d) f (h I k) z)

Задание 2.

№ варианта	Арифм. Операция	Список
1	+	(12 34 (5) 6)
2	-	((8) (1) 2)
3	+	(((6) 8) 32)
4	*	(3 (12 67) 2)
5	/	(100 (3) 4)
6	*	(4 3 (7))
7	*	((2 8) (4 5))
8	-	((90) 4 (23))

9	+	(2 (1) (5) 9)
10	/	(((30)) 2)
11	+	(10 (45 1))
12	*	(2 (3 4) 6 10)
13	-	(8 4 5 3)
14	/	(23 3 (4))
15	*	((12) (3) (5))
16	/	(120 (2 3) 4)

Задание 3.

№ варианта	Список	Условия проверки
1	(1 2 (3 4) 5 6 7)	а) список содержит всего два элемента; или б) список содержит три элемента.
2	(((q w e) g f) n)	а) первый и третий элемента списка одинаковы; или б) второй и третий элементы списка одинаковы
3	(a (x c) (s v) n q)	Хотя бы один из первых трех элементов является списком
4	((1 2 3) (4 5 6))	Первый элемент списка является атомом иначе Выдать сумму двух первых элементов первого элемента списка
5	(9 8 (7 6 5) e w)	первый или третий элемент списка является числом
6	((d) f (h I k) z)	Первый элемент списка равен второму или Второй элемент списка равен третьему

7	(3 (4 5) (j l))	второй или третий элементы списка являются списками
8	(h z ((q) d k l) a)	Первый элемент списка является числом или Третий элемент списка является списком
9	((1) (2) (3) (4) (5))	а) первый элемент списка равен 2; или б) второй элемент списка равен 3; или в) 3-й элемент списка = 4
10	(h ((a s d) l k)	Первый или третий элементы списка являются атомами
11	(0 (2 g a) 1 5)	Список содержит четыре элемента или Список пуст
12	((b) v (1 2 3) z a)	Второй или третий элементы списка являются списками
13	(e ((1) (2)) a s)	Один из первых трех элементов списка является числом
14	(r f b c n z)	Один из первых трех элементов списка является списком
15	((1) (3 4) 2 (5))	а) первый и третий элементы равны; или Б) первый и четвертый элементы равны; или В) второй элемент - список
16	(1 (2) в 4)	Если первый И второй И третий элементы - числа

2. ЛАБОРАТОРНАЯ РАБОТА 2. РАБОТА С ФУНКЦИЯМИ.

Цель работы.

Целью данной работы является знакомство с функциями и лямбда-выражениями в языке Лисп и получение первичных навыков написания собственных функций.

Методические указания.

Все задания можно выполнять в диалоговом режиме либо сначала записывать создаваемые функции в файл.

В случае записи функций в файл, их необходимо компилировать, и только после этого запускать выражения на выполнения.

Вызов созданных функций после компиляции можно осуществлять и в диалоговом режиме.

В процессе выполнения практики необходимо написать две функции, описанные в вашем варианте задания.

При выполнении второго пункта задания используйте функцию, описанную в первом пункте задания.

При выполнении обоих пунктов задания можно использовать, по необходимости, встроенные функции Лисп.

Задание на практику.

- a. записать функцию, определенную вашим вариантом.
- b. написать функцию, используя функцию, написанную в п.а)

Варианты заданий.

1. а) Напишите функцию, осуществляющую циклическую перестановку элементов в списке: первый элемент списка функция переставляет в конец списка, т.е. (a s d f) => (s d f a)

- б) написать функцию, которая сравнивает первые два элемента списка. Если эти элементы равны, возвращается исходный список, если разные, то первый элемент списка помещается в конец списка.
2. а) Определите функцию (f a b c), которая равна истине тогда и только тогда, когда из отрезков a, b и c можно построить треугольник.
 б) если из отрезков a, b и c можно построить треугольник, функция должна проверять, является ли треугольник прямоугольным и возвращать T или NIL; если треугольник построить нельзя, функция должна возвращать число, равное сумме трех чисел: a, b и c.
3. а) написать функцию, которая возвращает t, если первый и последний элементы списка равны, иначе - nil
 б) Определите функцию, которая меняет местами первый и последний элементы списка, оставляя остальные на своих местах, если элементы списка разные, иначе функция возвращает список без первого и последнего элементов.
4. а) Напишите функцию трех аргументов (list3 x y z), которая бы формировала список из аргументов: (x y z). Функцию list не использовать.
 б) написать функцию трех аргументов. Если все три аргумента атомы – создать новый список из этих аргументов. Если какой-то аргумент является списком, то в новый список включать только первый элемент списка:
 1 2 3 => (1 2 3)
 1 (a s) (d f c) => (1 a d)
5. а) Напишите функцию, единственный аргумент которой является списком. Функция возвращает t только в том случае, если первые два элемента этого списка являются списками. В остальных случаях функция должна выдавать nil.

- б) написать функцию с одним аргументом – списком. Если функция из п. а) возвращает t , создать список из четырех элементов: 1 и 2 элемент 1-го списка + 1 и 2 элементы 2-го списка. Если функция возвращает nil , то убрать из списка первые два элемента.
6. а) Написать функцию, на вход которой подается три целых числа – сторона квадрата, длина и ширина прямоугольника. Функция должна определять, сколько квадратов с заданной стороной поместится в этот прямоугольник (например, при стороне=2, ширине=3 и длине=5, можно поместить 2 квадрата).
- б) написать функцию, которая вычисляла бы общую площадь квадратов размером $s*s$, вписывающихся в прямоугольник размерами $a*b$.
7. а) написать функцию, которая выдает истину, если первые три элемента списка являются числами, и ложь в противном случае.
- б) создать функцию, на вход которой подается список Y и число N . Если первые три элемента списка Y являются числами, то функция возвращает список этих чисел, увеличенных в N раз. Если нет, то функция возвращает просто число N .
8. а) Напишите функцию двух аргументов ($f\ q\ w$), где аргументы являются списками. Функция возвращает t только в том случае, если первые два элемента этих списков соответственно равны друг другу. В остальных случаях функция должна выдавать nil .
- б) если предыдущая функция возвращает t , создать список из двух предыдущих путем слияния, исключив из них первые два элемента. Если nil – список из четырех элементов: 1 и 2 элемент 1-го списка + 1 и 2 элементы 2-го списка.
9. а) написать функцию от трех аргументов – списков. Функция должна возвращать 3, если в каждом списке

не менее 3-х элементов, 2 – если в каждом списке не менее 2-х элементов и 1 – если в каждом списке есть по крайней мере один элемент. Если среди исходных списков есть пустой – функция возвращает NIL.

б) если предыдущая функция возвращает i – создать список из i -х элементов исходных списков. Если NIL – вернуть пустой список.

10. а) написать функцию с одним аргументом – числовым списком, длиной не менее трех. Функция должна возвращать T, если первые три элемента списка – четные и NIL в противном случае.

б) если предыдущая функция возвращает T, вернуть исходный список, уменьшив его первые три элемента вдвое, иначе вернуть исходный список без первых трех элементов.

11. а) написать функцию с одним аргументом – списком.

Функция должна возвращать 2, если первые два элемента списка – списки, 1 – если только первый элемент является списком и 0 – в противном случае.

б) создать функцию, которая бы изменяла исходный список, поднимая первые элементы первых двух подсписков на верхний уровень, например:

$((a\ s)\ 2\ 3\ 4) \implies (a\ (s)\ 2\ 3\ 4)$

$((q\ w\ e)\ (1\ 2\ 3)\ 4\ 5\ (a)) \implies (q\ (w\ e)\ 1\ (2\ 3)\ 4\ 5\ (a))$

$(1\ 2\ (a\ 3)) \implies (1\ 2\ (a\ 3))$

12. а) написать функцию, которая меняла бы первые два элемента местами.

б) определить функцию, которая меняет местами элементы в исходном списке: 1-й и 2-й, 3-й и 4-й, 5-й и 6-й. Если количество элементов меньше 6, то менять местами 1-2, 3-4 (или только 1-2, если количество

элементов меньше 4-х). Для перестановки элементов использовать функцию из пункта а).

13. а) написать функцию с одним аргументом - списком, которая бы меняла местами последние два элемента списка.

б) определить функцию, на вход которой подается список и число. Если длина списка меньше заданного числа, то число добавляется в список в качестве первого элемента. Если больше, то функция меняет местами два последних элемента исходного списка.

14. а) определить функцию с двумя аргументами X и Y , где Y – список. Функция должна вставлять X в конец списка Y (не использовать функцию `append`).

б) написать функцию, имеющую два аргумента: X – S-выражение и Y – список. Если X равен первому элементу Y , то он добавляется в конец списка Y , если нет – в начало.

15. а) функция имеет три аргумента – действительные числа, которые являются параметрами квадратного уравнения. Функция должна возвращать `nil`, если уравнение не имеет корней и значение дискриминанта, если решение существует. Функция `SQRT` извлекает квадратный корень из аргумента.

б) определить функцию, которая решает квадратное уравнение, выдавая в качестве результата список из полученных корней.

3. ЛАБОРАТОРНАЯ РАБОТА 3. РЕКУРСИВНЫЕ ФУНКЦИИ.

Цель работы.

Целью данной работы является знакомство с рекурсией и принципами построения рекурсивных функций.

Методические указания.

Все задания можно выполнять в диалоговом режиме либо сначала записывать создаваемые функции в файл.

В случае записи функций в файл, их необходимо компилировать, и только после этого запускать выражения на выполнения.

Вызов созданных функций после компиляции можно осуществлять и в диалоговом режиме.

В процессе выполнения практики необходимо написать две функции, описанные в вашем варианте задания.

При написании функций сначала определите терминальную ветвь (при необходимости терминальных ветвей может быть несколько), которая определяет правило останова.

На втором этапе спланируйте рекурсивную ветвь (ветви), которая определяет рекурсивный вызов функции с упрощением аргументов.

Задание на практику.

Написать функции, заданные в вашем варианте с обязательным использованием рекурсии.

Варианты заданий.

1. а) напишите функцию, которая проверяет, являются ли все элементы списка положительными числами. Функция возвращает NIL, если хотя бы один элемент списка НЕ является числом, или если в списке есть хотя бы один отрицательный элемент.

- б) определите функцию умножения двух целых чисел через сложение и вычитание.
2. а) определите функцию, аргументом которой является числовой список z , и которая формирует новый список, удаляя из z на всех уровнях положительные элементы (z может быть многоуровневым).
 б) напишите функцию, которая увеличивает все элементы числового списка в два раза (список может быть многоуровневым).
3. а) определите функцию от двух аргументов x и n , которая создает список из n раз повторенных элементов x .
 б) напишите функцию, аналогичную встроенной функции `subst` замены в списке s выражения x на y , но производящую взаимную замену y на x : $x \rightarrow y, y \rightarrow x$.
 Например: $(f\ 2\ a\ (s\ d\ 2\ f\ a\ 4\ 2)) \implies (s\ d\ a\ f\ 2\ 4\ a)$
4. а) определите функцию, на вход которой подается список (он может быть многоуровневым). Функция должна возвращать `T`, если все атомы списка (на всех уровнях) являются числами, и `NIL` – в противном случае.
 б) определите функцию, на вход которой подается числовой одноуровневый список. В результате работы должен получиться числовой список, каждый i -й элемент которого является суммой первых i элементов исходного списка. Например: $(2\ 4\ 3\ 6\ 1\ 3) \implies (2\ 6\ 9\ 15\ 16\ 19)$
5. а) последовательность чисел Фибоначчи $1, 1, 2, 3, 5, 8, 13, \dots$ строится по следующему закону: первые два числа – единицы; любое следующее число

- есть сумма двух предыдущих. Напишите функцию (f n $f1$ $f2$) с накапливающимися параметрами $f1$ и $f2$, которая вычисляет n -е число Фибоначчи.
- б) напишите функцию, которая считает полное количество атомов (на всех уровнях и не равных NIL) во входном списке.
6. а) определите функцию, аргументом которой является список с четным количеством элементов – атомов. Функция должна менять местами каждую пару элементов списка (1-й элемент со 2-м, 3-й с 4-м и т.д.). Если количество элементов списка является нечетным, функция должна возвращать NIL.
- б) напишите функцию, имеющую два аргумента: числовой список и целое число N . Функция должна возвращать список, в котором элементы, стоящие на четных позициях, увеличены в N раз.
7. а) напишите функцию, аргументом которой является целое число, которая бы вычисляла факториал этого числа.
- б) напишите функцию, аргументом которой является список с любыми s -выражениями. Функция должна возвращать линейный список из всех атомов исходного. Например, список $(q (s d) (f (x c)) b)$ должен преобразовываться функцией в список $(q s d f x c b)$.
8. а) Написать функцию, на вход которой подается список из двух списков чисел. Результатом работы функции должен стать список, элементами которого являются поэлементные суммы этих двух подсписков. Например: $((1 3 2)(4 6 3)) \implies (5 9 5)$

б) Написать функцию, на вход которой подается список атомов. Функция должна формировать список из двух подсписков следующим образом: в 1-й подсписок включаются все элементы исходного списка, стоящие на нечетных позициях, во 2-й – на четных. Например: (ф ы в а п р) ==> ((ф в п)(ы а р))

9. а) Определите функцию, аргументом которой является одноуровневый список, элементами которого могут быть как числа, так и символьные атомы. Функция должна формировать новый список, состоящий из двух подсписков: в 1-й подсписок включаются числа, во 2-й – все символьные атомы.

Например: (в а 1 3 ы 5 ф ы) ==> ((1 3 5)(в а ы ф ы))

б) Напишите функцию, аргументом которой является список, удаляющую из этого списка все не числовые атомы на всех уровнях (список может быть многоуровневым).

10. а) Определите функцию, аргументом которой является многоуровневый список, атомы которого могут быть как числа, так и символьные атомы. Функция должна менять все числа на атом NUM.

Например: (а s 4 (s 1 2) с) ==> (а s num (s num num) с)

б) напишите функцию, определяющую глубину первого вхождения элемента у в список w.

11. а) определите функцию, аргументом которой является целое число n, результатом которой является сумма цифр натурального числа n. (встроенная функция MOD дает остаток от деления).

б) Определите функцию, аргументом которой является одноуровневый список. Функция должна менять все

отрицательные числа на атом MINUS, положительные – на атом PLUS, нули – на атом ZERO и символьные атомы – на атом SMV. Например: (a -4 s 1 0 c) ==> (smv minus smv plus zero smv)

12. а) Напишите функцию, которая делает из списка множество, т.е. удаляет все повторяющиеся элементы.
б) функция (f l) имеет единственный аргумент – список, состоящий из четного количества элементов – чисел. Каждая пара чисел в списке задает стороны прямоугольника. Функция должна возвращать список из периметров прямоугольников.
13. а) Определите функцию, реверсирующую список и все его подсписки на любом уровне. Например, (1 (2 3) 4) ==> (4 (3 2) 1).
б) Напишите функцию, зависящую от двух аргументов x и y, удаляющую все вхождения x в список y на всех уровнях.
14. а) напишите функцию, вычисляющую полное число подсписков, входящих в данный список на любом уровне. Например, для списка (a ((s) d f) (3 4)) оно равно трем.
б) на вход функции подается список, элементами которого являются списки из двух атомов. Функция должна менять местами элементы в каждом из подсписков: ((1 2) (ф ы) (3 в)) ==> ((2 1) (ы ф) (в 3))
15. а) напишите функцию, которая бы подсчитывала количество пустых списков на всех уровнях в единственном аргументе функции – списке.
б) На вход функции подается числовой список с четным количеством элементов. Определить функцию,

которая бы проводила расчеты по следующей схеме:
($x_1 x_2 x_3 x_4 \dots$) $\implies x_1 * x_2 + x_3 * x_4 + \dots$
Например: (1 2 3 4 5 6) $\implies 44$

16. а) Определить функцию, которая изменяет входной список, переставляя местами 1-й и 2-й элементы, 3-й и 4-й элементы и т.д.

Например: (a s d f g h) \implies (s a f d h g)

б) Определить функцию, на вход которой подается два числа K, L и числовой список, на выходе формируется список, положительные элементы которого увеличиваются в K раз, а отрицательные элементы увеличиваются в L раз. Например:

(2 3 (-1 2 3 -4 5)) \implies (-3 4 6 -12 10)

4. ЛАБОРАТОРНАЯ РАБОТА 4. РАЗРАБОТКА ФУНКЦИОНАЛЬНЫХ ПРОГРАММ.

Цель работы.

Целью данной работы является знакомство с рекурсией и принципами построения рекурсивных функций.

Методические указания.

В процессе выполнения практики необходимо написать две функции, описанные в вашем варианте задания.

Задание на практику.

Требуется составить программы на Лиспе, в каждом задании используя вспомогательные функции.

Варианты заданий.

1. Определите функцию, зависящую от одного аргумента, которая по данному списку формирует список его элементов, встречающихся в нем более одного раза.
2. Определите функцию $(f\ a\ n)$, которая от двух числовых аргументов вычисляет величину $a+a*(a+1)+a*(a+1)*(a+2)+\dots+a*(a+1)*(a+2)*\dots*(a+n)$.
3. Напишите функцию, на вход которой подается атом Y и список W . Функция должна заменять Y на число, равное глубине вложения Y в W , например, $Y=a$, $W=((a\ b)\ a\ (c\ (a\ (a\ d))))$ \rightarrow $((2\ b)\ 1\ (c\ (3\ (4\ d))))$.
4. Определите функцию $(f\ n)$ (n кратное 3), вычисляющую сумму:
 $1*2*3+4*5*6+\dots+(n-2)*(n-1)*n$.
5. Определите функцию $(f\ s)$, вычисляющую знакопеременную сумму $a_1-a_2+a_3-a_4+\dots+a_k*(-1)^{(k+1)}$ для списка s , имеющего вид $(a_1\ a_2\ a_3\ \dots\ a_k)$.
6. Определите функцию $(f\ n)$, которая для натурального числа n вычисляет $(1+0)!+(2+1)!+(3+2)!+\dots+(n+(n-1))!$
7. Определите функцию $(f\ s)$, которая вычисляет список $(m_1\ m_2\ m_3)$, состоящий из трех наибольших элементов числового списка s : $m_1 \geq m_2 \geq m_3$. Исходный список содержит не менее трех элементов.

8. Определите функцию $(f\ s)$, результатом которой является список, получающийся из списка списков s после удаления всех подсписков, содержащих числа.
9. Определите функцию $(f\ s)$, которая в одноуровневом списке s переставляет все отрицательные элементы в начало списка, например,
 $(f\ (4\ -2\ 3\ -1\ -9\ 8)) \rightarrow (-2\ -1\ -9\ 4\ 3\ 8)$
10. Определить функцию с одним аргументом – одноуровневым списком. Функция должна подсчитывать количество вхождений в список каждого атома и выдавать результат в виде списка списков: $(a\ s\ 3\ d\ s\ s\ a\ 3) \implies ((a\ 2)\ (s\ 3)\ (3\ 2)\ (d\ 1))$
11. Определите функцию, на вход которой подаются два списка – множества. Функция должна искать пересечение этих множеств.
12. Определите функцию, на вход которой подаются два списка – множества. Функция должна выдавать объединение этих множеств.
13. Определить функцию, которая определяет, является ли данное натуральное число простым.
14. Написать функцию, которая сортирует список чисел.
15. На вход функции подается одноуровневый список чисел. Необходимо найти номер первого экстремального (максимального или минимального) элемента списка.

16. определить функцию, которая находит максимальную сумму двух соседних чисел в заданном списке. Единственный аргумент функции является одноуровневым числовым списком.
17. Задан числовой список. Написать функцию, подсчитывающую среднее значение элементов списка, за исключением максимального и минимального элементов.

5. ЛАБОРАТОРНАЯ РАБОТА 5. ФУНКЦИОНАЛЫ.

Цель работы.

Целью данной работы является знакомство с функционалами и принципами их использования при написании новых функций.

Методические указания.

В процессе выполнения практики необходимо написать функцию с использованием функционала, описанную в вашем варианте задания.

Задание на практику.

Напишите функцию с использованием функционалов. При необходимости используйте локальные или вспомогательные функции.

1. Напишите функцию (`exist p x`), которая проверяет «Существует ли элемент списка `x`, удовлетворяющий предикату `p?`» (`p` – функция или функциональное имя).
2. Написать функцию, аргументом которой является числовой список. Функция должна возвращать список

сумм: 1-го элемента, 2-х первых элементов, 3-х первых элементов и т.д. Например: $(f \text{ `(1 2 3 4)}) \Rightarrow (1 3 6 10)$.

3. Определите функцию $(f \ s)$, которая из списка чисел s создает новый список, меняя знак у каждого атома. Исходный список не предполагается одноуровневым.
4. Напишите функцию $(\text{sum_list } s)$, аргументом которой является список арифметических выражений. Функция должна возвращать сумму вычисленных арифметических выражений. Например:
 $(\text{sum_list } \text{`((+ 1 2) (* 3 4) (- 8 4))}) \Rightarrow 19$
5. Напишите функцию $(\text{all } p \ x)$, которая проверяет «Для всех ли элементов списка x выполняется предикат p ?» (p – функция или функциональное имя).
6. Напишите функцию $(\text{count } p \ x)$, которая подсчитывает, сколько атомов в списке x удовлетворяет предикату p (p – функция или функциональное имя). Список x не предполагается одноуровневым.
7. Определите функцию $(f \ s \ n)$, которая из списка чисел s создает новый список, прибавляя к каждому атому число n . Исходный список не предполагается одноуровневым.
8. Напишите функцию $(\text{filter } p \ x)$, которая создает список из элементов списка x , удовлетворяющим предикату p (p – функция или функциональное имя).
9. Напишите функцию, строящую список всех подмножеств данного множества.

10. Напишите функцию ($f \text{ znak } l1 \ l2$), аргументами которой являются знак арифметической операции znak , и два числовых списка. Функция должна формировать новый список, состоящий из результатов применения арифметической операции попарно к элементам исходных списков. Например: $(f \text{ '+' } (1 \ 2 \ 3) \ (4 \ 5 \ 6)) \implies (5 \ 7 \ 9)$
11. Напишите функцию, на вход которой подается список числовых списков. Функция должна выдавать список, элементами которого являются максимумы исходных подсписков.
12. Напишите функцию ($\text{break } p \ x$), которая бы формировала список, состоящий из двух подсписков: в 1-й подсписок должны попадать все элементы x , удовлетворяющие предикату p , во 2-й – не удовлетворяющие.
13. Напишите функцию ($\text{oper } f \ y$), аргументами которой являются математическая функция f с тремя аргументами и числовой список y , длина которого кратна трем. Функция должна возвращать список результатов работы функции f над элементами списка y .
14. Напишите функцию ($f \ s \ n$), которая из списка чисел s создает новый список, исключив из него все элементы, которые превышают заданное число n .
15. Напишите функцию ($f\text{-}n \ p \ x \ n$), которая выдает истину только если ровно для n элементов списка x выполняется предикат p (p – функция или функциональное имя).
16. Напишите функцию ($f \ n \ l_f \ x$), где n – число, l_f – список арифметических операций, x – числовой список. Функция

должна возвращать число – результат применения n-й операции к элементам списка x.

6. ЛАБОРАТОРНАЯ РАБОТА 6. ЦИКЛЫ И БЛОЧНЫЕ ФУНКЦИИ.

Цель работы.

Целью данной работы является знакомство с циклами и блочными функциями и принципами их использования при написании новых функций.

Методические указания.

В процессе выполнения практики необходимо написать функцию с использованием циклов и блочных функций, описанную в вашем варианте задания.

Задание на практику.

Напишите функцию с использованием циклов и блочных функций. При необходимости используйте локальные или вспомогательные функции.

Внимание! Рекурсию не использовать!

В качестве вариантов заданий используйте свои варианты из работы 3: Рекурсивные функции.

7. ЛАБОРАТОРНАЯ РАБОТА 7. ГРАФЫ И ДЕРЕВЬЯ.

Цель работы.

Целью данной работы является работа с графами, представленными различными способами и написание функций для работы с ними.

Методические указания.

В процессе выполнения практики необходимо написать

функцию работы с графом (деревом), выполняющую заданное действие.

Задание на практику.

Напишите функцию, определяющую для графа (дерева) заданную характеристику. При необходимости используйте локальные или вспомогательные функции.

Варианты заданий.

1. Дан граф. Написать функцию, которая находит в графе максимальный цикл и выдает его в виде списка вершин. Если в графе нет циклов, функция должна возвращать nil.
2. Написать функцию, на вход которой подается граф, определяющую, связан ли граф.
3. Задан граф, у которого для каждой дуги задана ее длина: ((a b 12) (s d 3) ...). Написать функцию, определяющую кратчайший путь между указанными двумя вершинами.
4. Написать функцию, подсчитывающую количество циклов в графе.
5. Определить функцию, аргументом которой является дерево, подсчитывающую количество листьев в дереве.
6. На вход функции подается ориентированный граф. Функция должна определять, существует ли путь из A в B.
7. Написать функцию, на вход которой подается дерево, определяющую максимальную глубину дерева.
8. Написать функцию, на вход которой подается граф в виде (a (b) b (a c d) c (b e f g) ...). Функция должна преобразовывать граф в вид: ((a b) (b c) (b d) ...).
9. Написать функцию, на вход которой подается граф в виде (a (b) b (c d) c (e f g) ...). Функция должна проверять, является ли граф ориентированным.
10. Определите функцию, аргументом которой является дерево. Функция должна вернуть ветвь с максимальным количеством листьев.

11. Определить функцию, которая ищет заданную вершину в дереве и возвращает список, содержащий предка искомой вершины и ее потомков: (предок (потомок1 потомок2 ...)).
12. Задан граф, у которого для каждой дуги задана ее длина: ((a b 12) (s d 3) ...). Определить функцию, которая находит две самые удаленные друг от друга вершины.
13. Написать функцию, которая проверяет, является ли заданный граф деревом (имеет ли циклы)?
14. На вход функции подается дерево. Функция должна добавить к каждой вершине информацию: глубину ветки, исходящей из данной вершины. Например, дерево вида (1 (2 (3 (4))(5)) (6 (7))) должно преобразоваться в ((1 3) ((2 2) ((3 1) ((4 0))((5 0))) ((6 1) ((7 0))))
15. Определить функцию, которая ищет середину кратчайшего пути между двумя заданными вершинами. Функция должна возвращать: атом-имя вершины (если между исходными вершинами нечетное количество вершин), список из двух вершин (если четное) и NIL, если заданные вершины – соседи.
16. Стяжение ветви. Определить функцию, аргументами которой является дерево и две его вершины. Функция должна склеивать два заданных узла, если они соседние и выдавать NIL в противном случае.

8. КУРСОВАЯ РАБОТА.

Цель работы.

Целью данной работы является знакомство с моделями представления знаний: фреймами и семантическими сетями, а также получение практических навыков создания Баз Знаний и работы с

ними.

Методические указания.

В процессе выполнения курсовой работы необходимо создать комплекс программ, позволяющий создавать и корректировать базу знаний, а также выполнять поиск требуемой информации в созданной базе.

Задан список предметных областей для создания Базы Знаний. Выберите предметную область из предложенного списка (можно предложить собственную предметную область и согласовать ее с преподавателем).

Выберите модель представления знаний: фреймы или семантическую сеть.

При выборе фреймов необходимо построить фреймовую структуру в заданной области: построить иерархическую классификацию и отобразить ее с помощью языка Лисп. Хотя бы один из фреймов должен содержать применяющую процедуру.

При выборе семантической сети необходимо построить ориентированный граф, определить объекты и отношения в сети; реализовать сеть с помощью языка Лисп.

Для построения можно пользоваться фрагментами программ (программами) рассмотренными на лекции.

Создать интерфейс для ввода и отображения информации.

Реализовать функцию работы с Базой Знаний, определенную преподавателем.

Задание на выполнение.

1. Выбрать предметную область.
2. Выбрать модель представления знаний (фреймы, семантическая сеть).
3. Для фреймовой модели: определить структуру фрейма; построить иерархическую классификацию; определить хотя бы одну применяющую процедуру.
4. Для семантической сети: определить объекты сети и возможные отношения между ними; построить граф сети; определить возможные вопросы, на которые должна отвечать система.

5. Реализовать разработанную модель на языке Лисп.

Возможные функции:

- a. Удаление фрейма из структуры:
 - удаление имени фрейма из списка дочерних фреймов предка;
 - удаление всех свойств фрейма;
 - удаление всех дочерних фреймов удаляемого фрейма.
- b. Удаление объекта из сети: удаление имени объекта; удаление всех связей объекта; удаление всех объектов, связанных с удаляемым отношениями IS-A.
- c. изменение информации о фрейме (любой слот может быть изменен).
- d. просмотр информации о фрейме: Выводить предка фрейма, список дочерних фреймов, свойства фрейма.
- e. поиск всех фреймов с заданным свойством (в качестве поискового значения может быть задано значение любого свойства фрейма или имя слота).
- f. поиск фрейма с заданными свойствами (не обязательно полный список свойств, свойства могут быть заданы в различном порядке).
- g. Вывод всех свойств заданного фрейма (включая свойства всех предков)
- h. Поиск всех фреймов-экземпляров с заданным свойством (конечных фреймов на дереве иерархии)
- i. Вывод всей фреймовой иерархии
- j. добавление информации о фрейме (слот может быть добавлен или удален)
- k. определение всех предков заданного фрейма (отца, деда, прадеда и т.д.)

- l. определение всех потомков заданного фрейма (детей, внуков, и т.д.)
- m. изменение имени заданного фрейма (учесть, что он может быть для других фреймов как отцом, так и потомком)

Список предметных областей:

1. цветы (ботаника).
2. дисциплины в учебном плане. Примечание: здесь каждый слот для конечного фрейма – конкретной дисциплины (лекции, практики, лаб. раб, курс. раб., самост. раб.) имеет числовое значение (кол. часов).
3. книги.
4. деревья.
5. автомобили.
6. деревья.
7. фильмы.
8. птицы.
9. насекомые.
10. инструмент.
11. строения.
12. рыбы (водный мир)
13. дом/жилище

9. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ВЫПОЛНЕНИЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

9.1. Цели самостоятельной работы

Цель самостоятельной работы по дисциплине – проработка лекционного материала, самостоятельное изучение некоторых разделов курса, подготовка к лабораторным работам, опросам на лекциях и контрольным работам.

9.2. Содержание самостоятельной работы

9.2.1. Проработка лекционного материала

Содержание

Концепция и особенности функционального программирования. Свойства функциональных языков. Основные особенности Лиспа, достоинства языка.

Элементарные понятия языка Лисп: атомы и списки. Программа на языке Лисп. Вычислимые выражения. Понятие функции, префиксная нотация.

Вычисление лямбда-выражений. Порядок редукций и нормальные формы. Определение функций в Лиспе. Базовые функции языка, предикаты.

Понятие рекурсии. Правила записи рекурсивной функции. Терминальная ветвь, рекурсивная ветвь. Прямая и косвенная рекурсия. Рекурсия с несколькими терминальными ветвями, рекурсивными ветвями.

Внутреннее представление списков. Вспомогательные функции над списками.

Глобальные и локальные переменные. Изменение значений переменных. Диалоговый режим работы. Функции ввода-вывода. Циклы и блочные функции. Обработка текстовых данных. Работа с файлами. Массивы.

Функции высших порядков. Различие между данными и функциями. Функционалы.

Обработка и хранение знаний: свойства символов; ассоциативные списки.

9.2.2. Темы для самостоятельной проработки

Каждый студент должен самостоятельно изучить следующие темы, вопросы по которым будут включаться в экзаменационные билеты.

- 1) Разрушающие функции. Обратная блокировка.
- 2) Деревья

10. СПИСОК ЛИТЕРАТУРЫ

1. Салмина Н. Ю., Функциональное программирование и интеллектуальные системы: учебное пособие [Электронный ресурс] / Салмина Н. Ю. — Томск: ТУСУР, 2016 . — 100 с. — Режим доступа: <https://edu.tusur.ru/publications/6357>.

2. Кубенский А.А. Функциональное программирование: Учебное пособие. – С-Птб.:

СПбНИУ ИТМО, 2010. – 251 стр. (ГРИФ) – ЭБС ЛАНЬ –

http://e.lanbook.com/books/element.php?p11_id=40771

3. Роганова, Н. А. Функциональное программирование : учебное пособие для вузов / Н. А. Роганова ; Федеральное агентство по образованию, Московский государственный индустриальный университет. - М. : МГИУ, 2007. - 214 с.

4. Зюзьков, Валентин Михайлович. Функциональное программирование : учебное пособие / В. М. Зюзьков ; Федеральное агентство по образованию, Томский государственный университет систем управления и радиоэлектроники, Кафедра автоматизированных систем управления. - Томск : ТМЦО, 2005. - 140 с.