Министерство науки и высшего образования РФ

Севастопольский государственный университет

Кафедра информатики и управления в технических системах

ОТЧЕТ

по лабораторной работе №7

Разработка и исследование рекурсивных функций на Лиспе

по дисциплине «Логическое программирование»

Выполнил:

Студент группы ИВТ/б 22-о

Черняев Н.Г.

Проверил:

Ткаченко К.С.

г. Севастополь 2019

**Цель работы**

Изучить основы программирования с применением рекурсии.

**Вариант № 0,4,11,18**

В примерах для выполнения самостоятельной работы требуется исследовать схему рекурсивного процесса, подобрать корректные входные данные для функции. Описать процесс выполнения рекурсивных функций.

Задание к варианту 0:

(defun F0(a b c)

(F1 (\* 4 a c) (\* b b))

)

(defun F1(d e)

(F2 (+ d e))

)

(setq g 2.0)

(setq f 100.0)

(defun F2(f)

(F3 (sqrt f) g)

)

(setq b 10.0)

(defun F3(b g)

(F4 (+ (- b) g) g)

(F4 (+ (- b) g) g)

)

(setq h 10.0)

(setq s 2.0)

(defun F4(h s)

(let ((h 7) (s 3))(print(/ h (\* 2 s))))

)

(let ((a1 7) (b1 3))(\* a1 b1))

(defun FR1(a1 b1 a2 b2 a3 b3 a4 b4)

(F2 (\* (+ a1 b1) (- a2 b2) (/ a3 b3) (\* a4 b4)))

)

(defun FR2(x y z m c d)

(cond

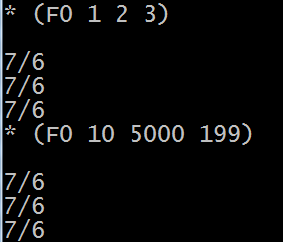
((> y 0) (F4 (\* x m) (+ c d)))

(T (F3 (- c d) (+ x z)))

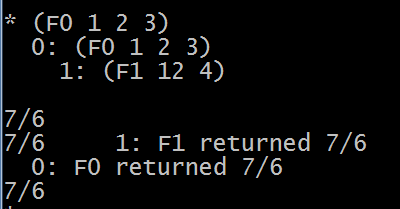
)

)

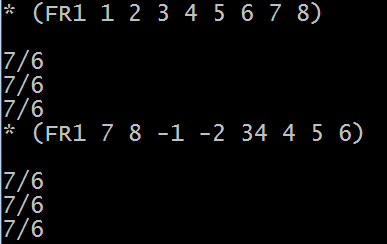
Результат работы функции F0.



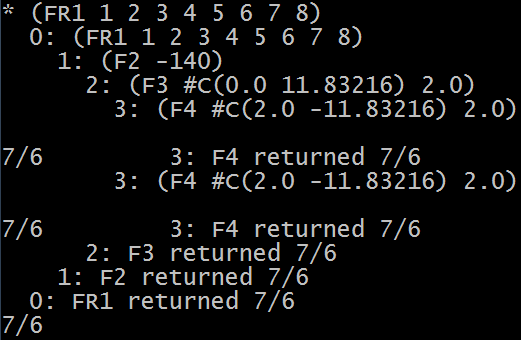
Результат трассировки функции:



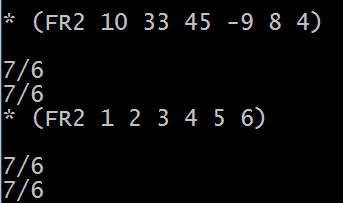
Результат работы функции FR1.



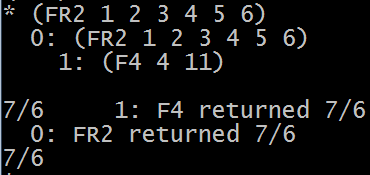
Результат трассировки функции:



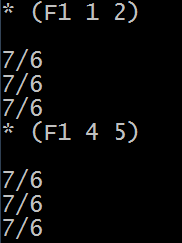
Результат работы функции FR2.



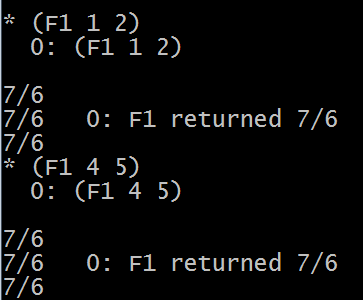
Результат трассировки функции:



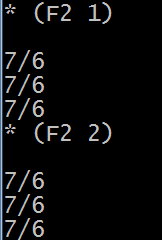
Результат работы функции F1.



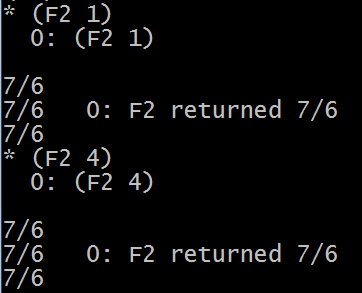
Результат трассировки функции:



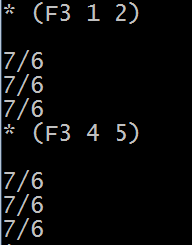
Результат работы функции F2.



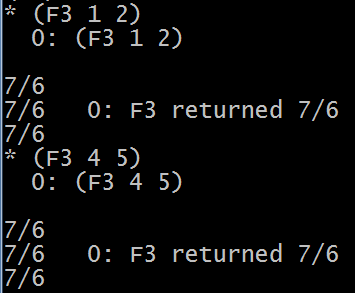
Результат трассировки функции:



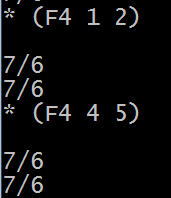
Результат работы функции F3.



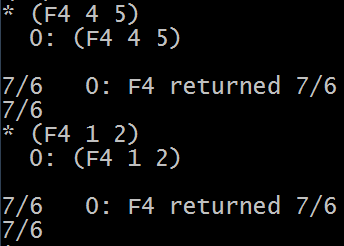
Результат трассировки функции:



Результат работы функции F4.



Результат трассировки функции:



Исходя из результатов трассировки функции FR1 и FR2 – рекурсивные.

Задание к варианту 4:

(defun SUM(Lst)

(COND

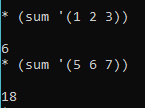
((NULL Lst) 0)

(T (+ (CAR Lst) (SUM (CDR Lst)) ))

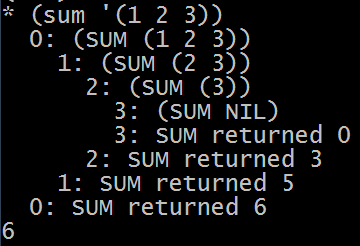
))

Функция SUM суммирует элементы списка.

Пример работы функции:



Результат трассировки функции:



Задание к варианту 11:

(defun SFR(Rev Lst)

(COND

((NULL Lst) Rev)

(T (SFR(cons (CAR Lst)Rev) (CDR Lst)))))

(defun FR(Lst)

(SFR nil Lst)

)

(defun Ins(In Lst)

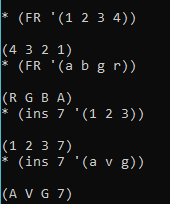
(COND ((NULL Lst)(List In))

(T (cons (CAR Lst)(Ins In (CDR Lst))))

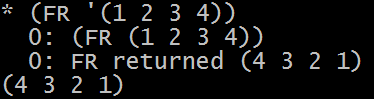
))

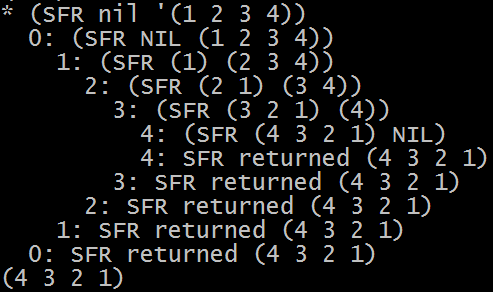
Функция FR делает реверс спика, а функция INS добавляет элмент в конец списка.

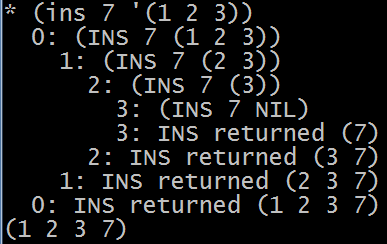
Пример работы функций:



Результат трассировки функций:







Задание к варианту 18:

(defun GLst (E1 Lst)

(COND

((NULL Lst) NIL)

((<= E1 (CAR Lst)) (GLst E1 (CDR Lst)))

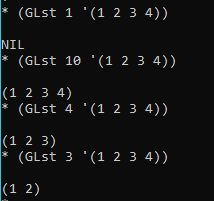
(T (CONS (CAR Lst) (GLst E1 (CDR Lst))))

)

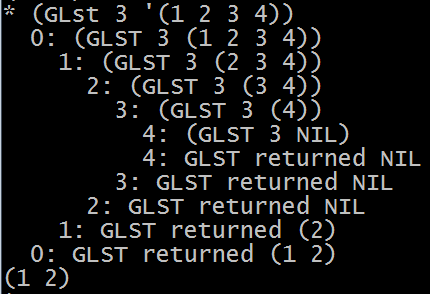
)

Данная функция удаляет все элементы из спика Lst, которые больше или равны Е1.

Пример работы:



Результат трассировки функций:



**Выводы**

В данной лабораторной работе были изучены основы программирования с применением рекурсии.