



**Министерство науки и высшего образования Российской
Федерации
Федеральное государственное бюджетное образовательное
учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

**Отчёт
к лабораторной работе № 2
По курсу: «Функциональное и логическое программирование»
Тема: «Списки в Lisp. Использование стандартных функций.»**

Студент Прохорова Л. А.

Группа ИУ7-63Б

Оценка (баллы) _____

Преподаватели Толпинская Н. Б., Строганов Ю. В.

Москва.
2021 г.

Цель работы: приобрести навыки использования списков и стандартных функций Lisp.

Задачи работы: изучить внутреннее представление одноуровневых и структурированных списков и методы их обработки с использованием базовых функций Lisp.

Задание 2

Используя только функции car и cdr, написать выражения, возвращающие

1) второй

2) третий

3) четвертый элементы заданного списка

1) (car (cdr '(A B S D E)))

(cdr '(A B S D E)) вернёт (B S D E)

(car (cdr '(A B S D E))) вернёт B

2) (car (cdr (cdr '(A B S D E))))

3) (car (cdr (cdr (cdr '(A B S D E)))))

Задание 3

Что будет в результате вычисления выражений?

1) (caadr '((blue cude)(red pyramid))) Результат red

Запись (caadr X) эквивалентна (car (car (cdr X))).

(cdr '((blue cude)(red pyramid))) -> ((red pyramid))

(car '((red pyramid))) -> (red pyramid)

(car '(red pyramid)) -> red

2) (cdar '((abc) (def) (ghi))) Результат NIL

Запись (cdar X) эквивалентна (cdr (car X)).

(car '((abc) (def) (ghi))) -> (abc)

(cdr '(abc)) -> Nil

3) (cadr '((abc)(def)(ghi))) Результат (def)

Запись (cadr X) эквивалентна (car (cdr X))

(cdr '((abc)(def)(ghi))) -> ((def)(ghi))

(car ' ((def)(ghi))) -> (def)

4) (caddr '((abc)(def)(ghi))) Результат (ghi)

Запись (caddr X) эквивалентна (car (cdr (cdr X)))

```
(cdr '((abc)(def)(ghi))) -> ((def)(ghi))  
(cdr '(((def)(ghi)))) -> ((ghi))  
(car '((ghi))) -> (ghi)
```

Задание 4

Найти результат вычисления выражений

1) (list 'Fred 'and Wilma)

Результат: Error (The variable WILMA is unbound.)

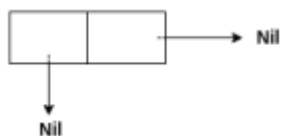
Ошибка так как при Wilma нет апострофа и Lisp считает надо исполнить Wilma, но такой переменной нет.

2) (list 'Fred '(and Wilma))

Результат: (FRED (AND WILMA))

3) (cons Nil Nil)

Результат: (NIL)

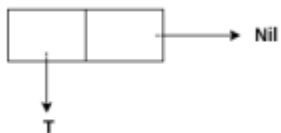


cons возвращает точечную пару обе ячейки которой содержат Nil - это интерпретируется как список, голова которого содержит Nil, а хвост пустой.

4) (cons T Nil)

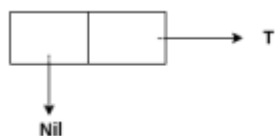
Результат: (T)

cons возвращает точечную пару ячейки которой содержат T и Nil - это интерпретируется как список, голова которого содержит T, а хвост пустой.



5) (cons Nil T)

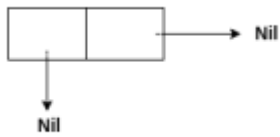
Результат: (NIL . T)



6) (list Nil)

Результат: (NIL)

Списковая ячейка голова которой содержит значение Nil, а хвост пустой.



7) (cons (T) Nil)

Результат: Error (The function T is undefined)

(T) - интерпретируется как вызов функции T без параметров, такой функции нет.

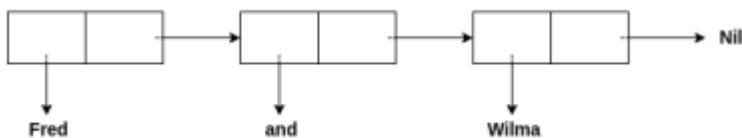
8) (list '(one two) '(free temp))

Результат: ((ONE TWO) (FREE TEMP))

9) (cons 'Fred '(and Wilma))

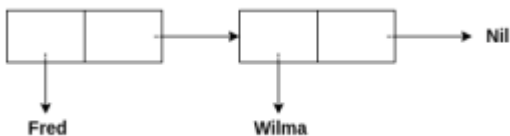
Результат: (FRED AND WILMA)

Создана точечная пара в одной ячейке которой Fred, а вторая ячейка указывает на двухэлементный список (and Wilma). В памяти это представляется аналогично трехэлементному списку.



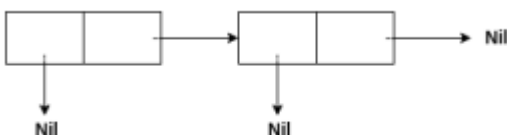
10) (cons 'Fred '(Wilma))

Результат: (FRED WILMA)



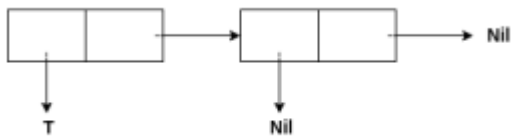
11) (list Nil Nil)

Результат: (NIL NIL)



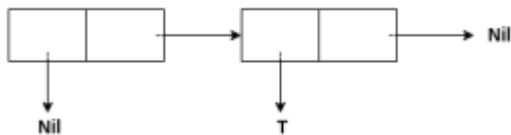
12) (list T Nil)

Результат: (T NIL)



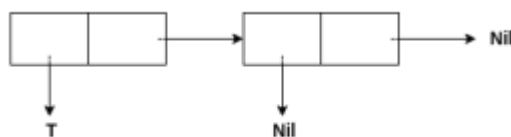
13) (list Nil T)

Результат: (NIL T)



14) (cons T (list Nil))

Результат: (T NIL)

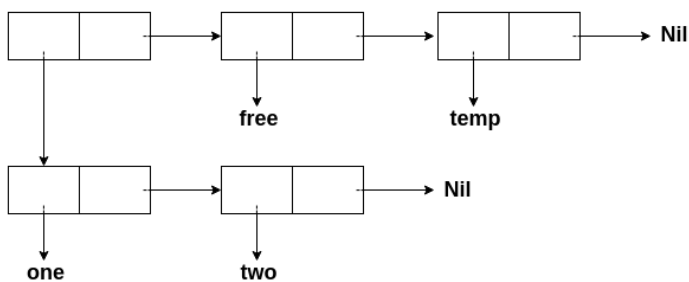


15) (list (T) Nil)

Результат: Error (The function T is undefined)

16) (cons '(one two) '(free temp))

Результат: ((ONE TWO) FREE TEMP)

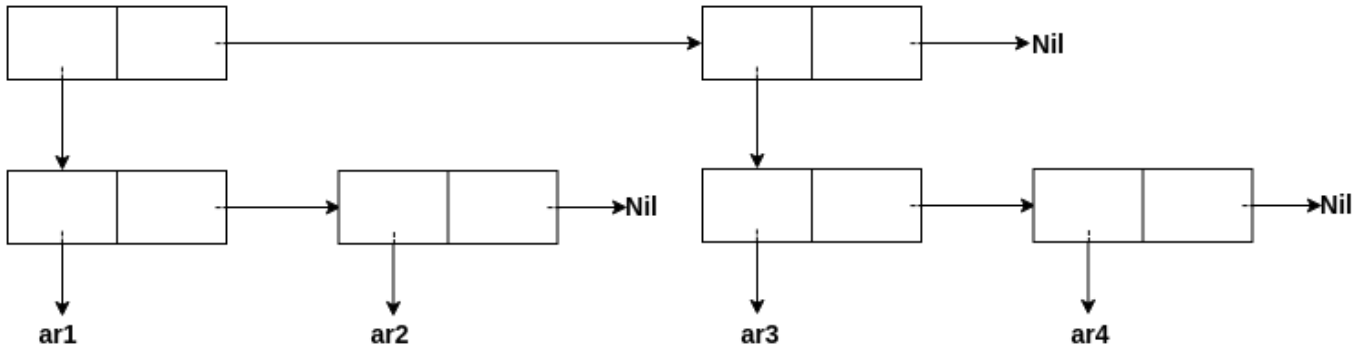


Задание 5

Написать функцию (f ar1 ar2 ar3 ar4), возвращающую список: ((ar1 ar2)(ar3 ar4)).

(defun f1 (ar1 ar2 ar3 ar4) (list (list ar1 ar2) (list ar3 ar4)))

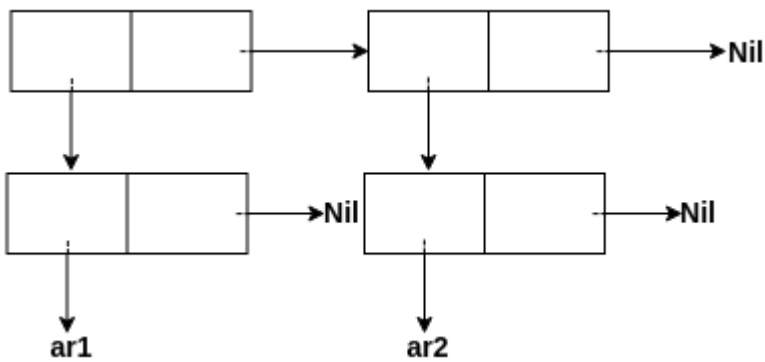
(f1 3 5 2 1) Результат: ((3 5)(2 1))



Написать функцию (f ar1 ar2), возвращающую ((ar1)(ar2))

(defun f2 (ar1 ar2) (list (list ar1) (list ar2)))

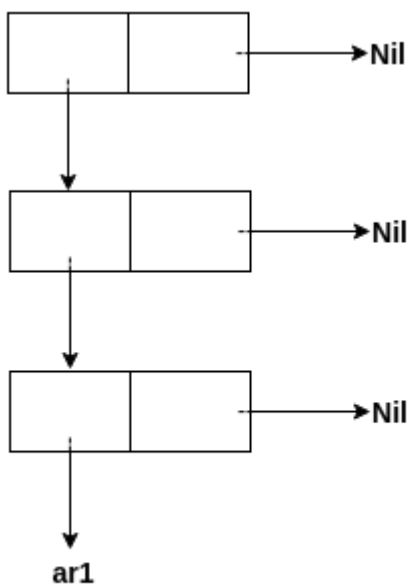
(f2 3 'a) Результат: ((3) (A))



Написать функцию (f ar1), возвращающую (((ar1)))

(defun f3 (ar1) (list (list (list ar1))))

(f3 4) Результат: (((4)))



Ответы на теоретические вопросы:

1) Классификация функций в Lisp

Функции с точки зрения организации

- Чистые математические функции (фиксированное количество элементов и определён результат)
- Рекурсивные функции
- Специальные функции (произвольное количество аргументов или по разному обрабатывают свои аргументы)
- Псевдофункции (создают эффект на экране)
- Функции с вариантами значений из которых возвращается только одно значение
- Функции высших порядков (Используются для построения синтаксически управляемых программ. Это абстракция языка. Если описание одной функции синтаксически похоже на описание другой, то вместо двух похожих пишется одна)

Функции с точки зрения действий

- Конструкторы (Пример: cons, list)
- Селекторы (Пример: car, cdr)
- Предикаты (в математике это логические функции, для Lisp используется принцип: то что не Nil, является true) Пример: eq

2) Базис Lisp

- Атомы
- Бинарные узлы (позволяют организовать точечные пары или списки)
- Функции: atom, eq, cons, car, cdr
- Функции: cond, quote, lambda, eval, label

3) Как выполняются функции car и cdr

CAR и CDR - базовые функции доступа к данным. CDR принимает точечную пару или пустой список и возвращает список состоящий из всех элементов, кроме первого. Если в списке меньше двух элементов, то возвращается Nil. CAR принимает точечную пару или пустой список в качестве аргумента и возвращает первый элемент или nil, соответственно.

4) В чем отличие выполнения функций list и cons?

LIST и CONS используются для создания списков.

CONS работает эффективно. Эта функция создает одну списковую ячейку и расставляет 2 указателя на аргументы. Входит в базис.

LIST входит в ядро. Ядро - это основные действия которые наиболее часто используются. Эта функция принимает переменное число аргументов и возвращает список, элементы которого - переданные в функцию элементы. Эта функция написана с использованием функции CONS.