

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»

Лабораторная работа №1
по курсу: «Функциональное и логическое
программирование»
**Списки в Lisp. Использование стандартных
функций**

Студент: Набиев Ф.М.

Группа: ИУ7-63Б

Преподаватель: Толпинская Н.Б.

2020 г.

ВВЕДЕНИЕ

Целью работы является приобретение навыков использования списков и стандартных функций Lisp.

Задачи работы: изучить способ использования списков для фиксации информации, внутреннее представление одноуровневых и структурированных списков, методы их обработки с использованием базовых функций Lisp.

1 Теоретические сведения

Базис в Lisp образуют атомы, структуры, базовые функции, базовые функционалы.

Вся информация в Lisp представляется в виде S-выражений. **S-выражения** - это атом или точечная пара. **Атомами** являются:

- **символы** - идентификаторы, набор букв и цифр, начинающийся с буквы;
- **T, Nil** - специальные символы, использующиеся для обозначения логических констант. Кроме того, Nil означает пустой список, а все значения, не равные Nil, интерпретируются как T;
- **самоопределяемые атомы** - натуральные числа, дробные числа, вещественные числа, строки.

Так же существуют более сложные структуры данных - списки и точечные пары.

Точечной парой является конструкция вида (A . B), где под A и B подразумевается либо атом, либо точечная пара.

Список быть пустым или не пустым. **Пустой список** - это () или, как было упомянуто выше, Nil. **Не пустой список** - это точечная пара, состоящая из головы и хвоста. **Голова списка** - это S-выражение, а **хвост** - список.

2 Практическая часть

2.1 Задание №1

Представления списков, указанных в условии данной лабораторной работы, в виде списочных ячеек изображены на рисунках 2.1-2.6.

а) '(open close halph)

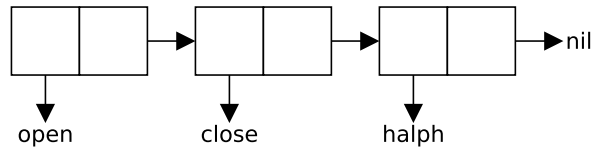


Рисунок 2.1 — Список '(open close halph)

б) '((TOOL) (call))

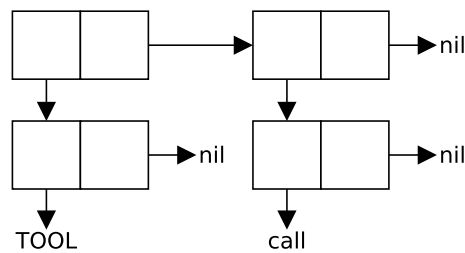


Рисунок 2.2 — Список '((TOOL) (call))

в) '((open1) (close2) (halph3))

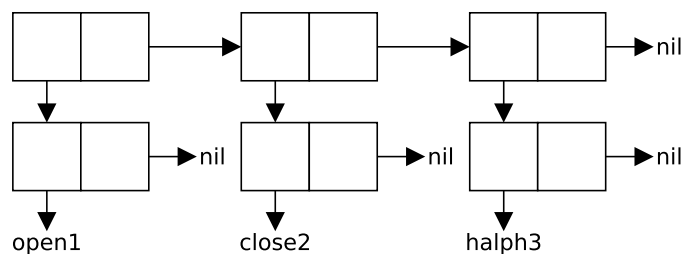


Рисунок 2.3 — Список '((open1) (close2) (halph3))

г) '((TOOL1) ((call2)) ((sell)))

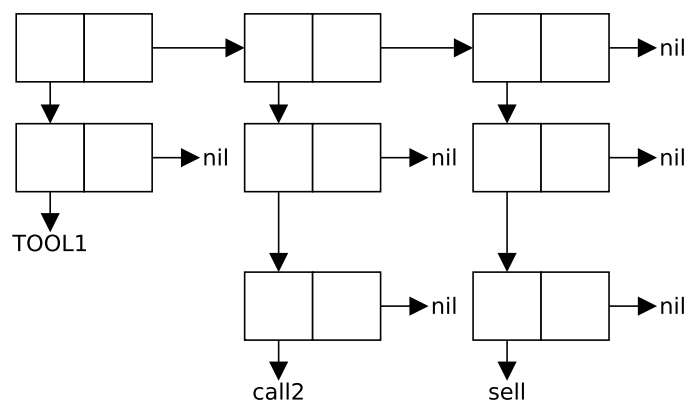


Рисунок 2.4 — Список '((TOOL1) ((call2)) ((sell)))

д) '(((TOOL) (call)) ((sell)))

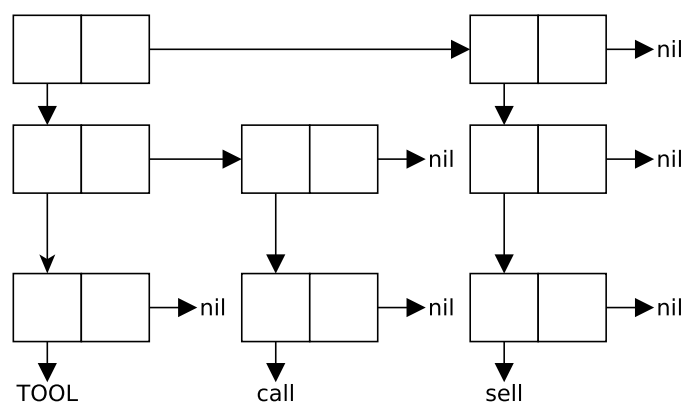


Рисунок 2.5 — Список '(((TOOL) (call)) ((sell)))

е) '((one) for all (and (me (for you))))

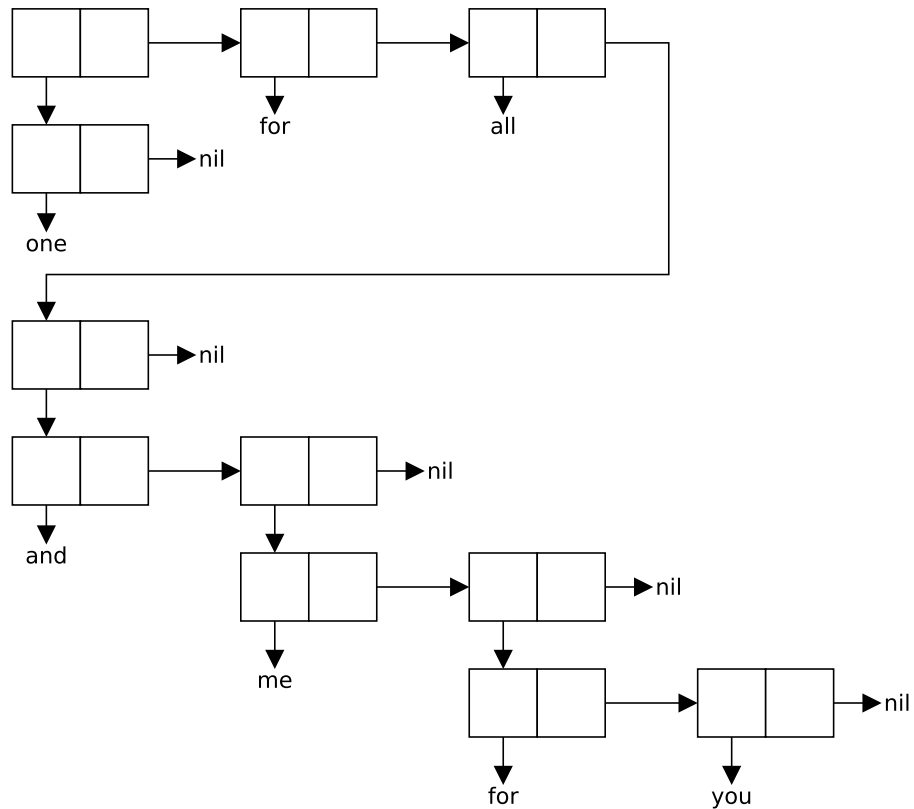


Рисунок 2.6 — Список '((one) for all (and (me (for you))))

2.2 Задание №2

В листинге 2.1 приведены три выражения на языке Lisp, которые возвращают второй, третий и четвёртый элементы списков. Для этого были использованы функции доступа `car` и `cdr`. Функция **`car`** возвращает голову списка, а **`cdr`** - его хвост. Результаты выражений приведены в комментариях под соответствующим им строкам.

Листинг 2.1 — Выражения, возвращающие 2, 3 и 4 элемента списка

```
1 (car (cdr '((((1)))) (2 3 4 5) (3 (3 (3 (4)))) 4 (8 9))))
2 ; (2 3 4 5)
3 (car (cdr (cdr '((3 41 2) (2 (3)) (5 6 (6 9) 8) (7 9 3 2 (4 2 (3))) 5))))
4 ; (5 6 (6 9) 8)
5 (car (cdr (cdr (cdr '(1 (2 2 3 4 5) ((3) 7 8) (4 9 7 (((7)))) 5)))))
6 ; (4 9 7 (((7))))
```