|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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 предмету: «Функциональное и логическое программирование»*

**Тема:** Списки в Lisp. Использование стандартных функций.

Студент: Ятагани. К.,

Группа: ИУ7-66И

Москва, 2020 г.

Оглавление

[Введение 3](#_Toc33126324)

[Методологии программирования: 3](#_Toc33126325)

[Основные идеи функционального программирования: 3](#_Toc33126326)

[Задание №1 4](#_Toc33126327)

[Задание №2 6](#_Toc33126328)

[Ответы на вопросы 7](#_Toc33126329)

## Введение

## Методологии программирования:

* Императивное
* Функциональное
* Логическое
* Объектно-ориентированное

## Основные идеи функционального программирования:

* Отсутствие понятие «оператор»
* Базовое понятие – «функция»
* Базовая структура – «список»
* Нет четкого разделения на программу и данные

Функциональное (аппликативное) программирование - базовым понятием является функция (математическое понятие); за состоянием памяти следить не нужно - важен результат (возврат), а не процесс (состояние памяти). Операторов и команд нет в принципе. Функциональное программирование ориентировано на символьную обработку данных. Предполагается, что любую информацию можно свести к символьной. Слово «символ» здесь близко к понятию «знак».

Базис Lisp образуют:

атомы, структуры, базовые функции, базовые функционалы.

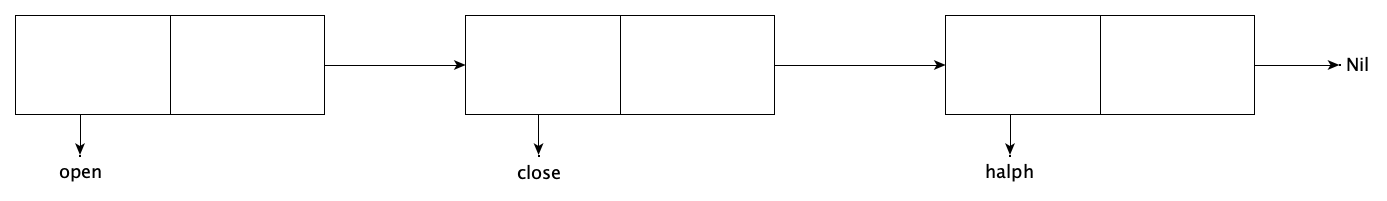
**Цель работы:** приобрести навыки использования списков и стандартных функций Lisp.

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

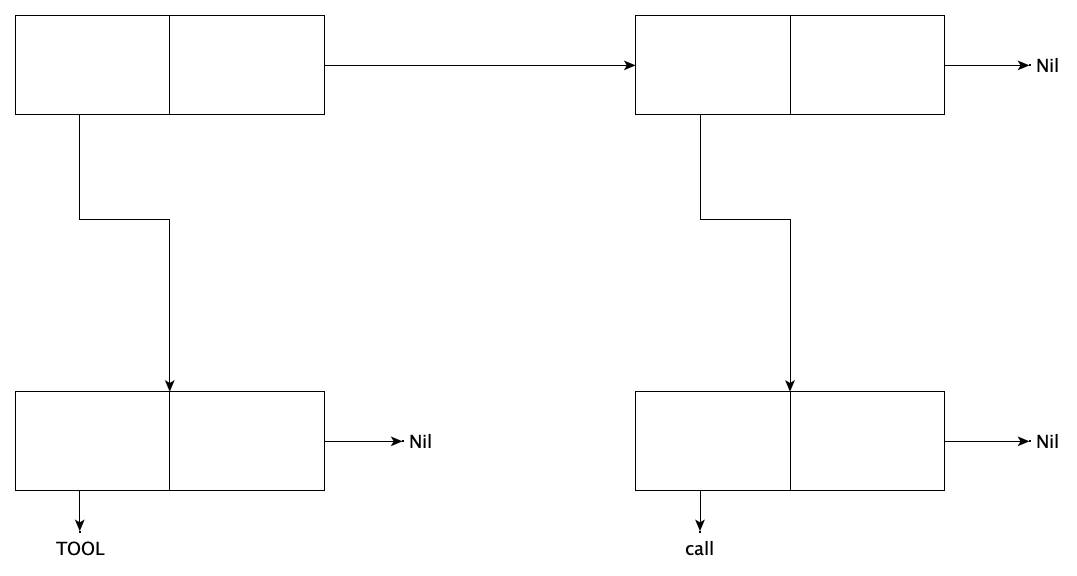
## Задание №1

1. Представить следующие списки в виде списочные ячеек:

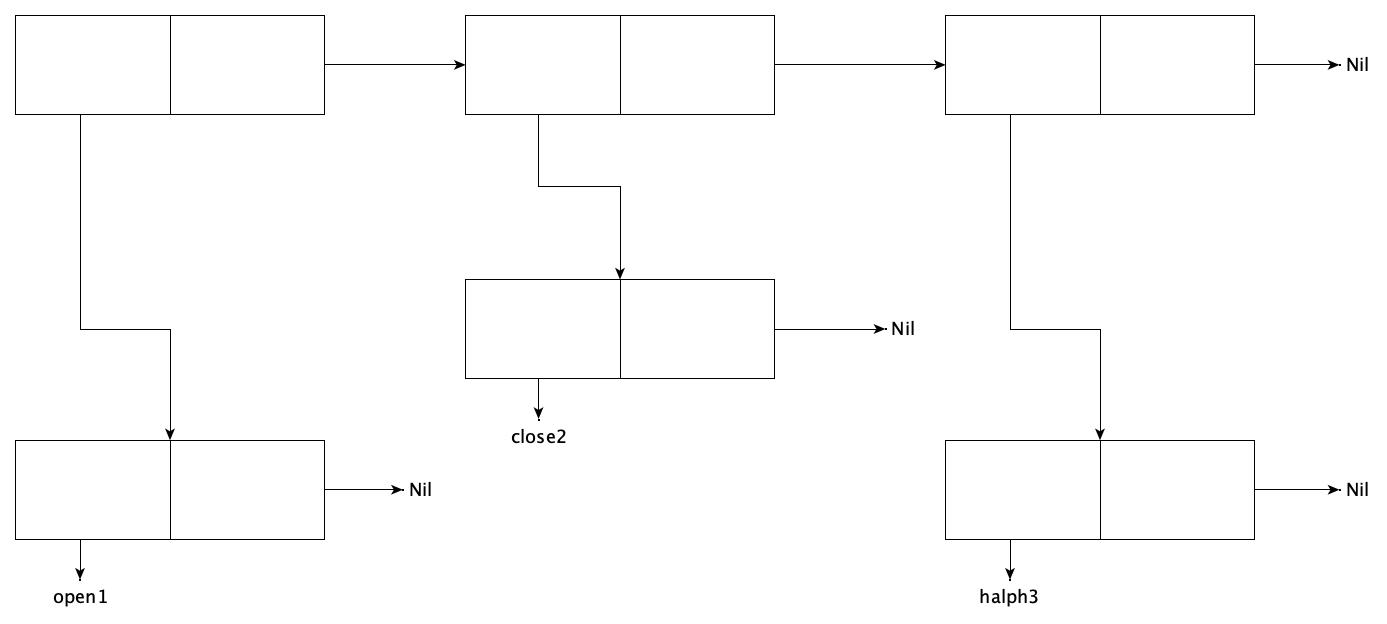
' (open close halph)



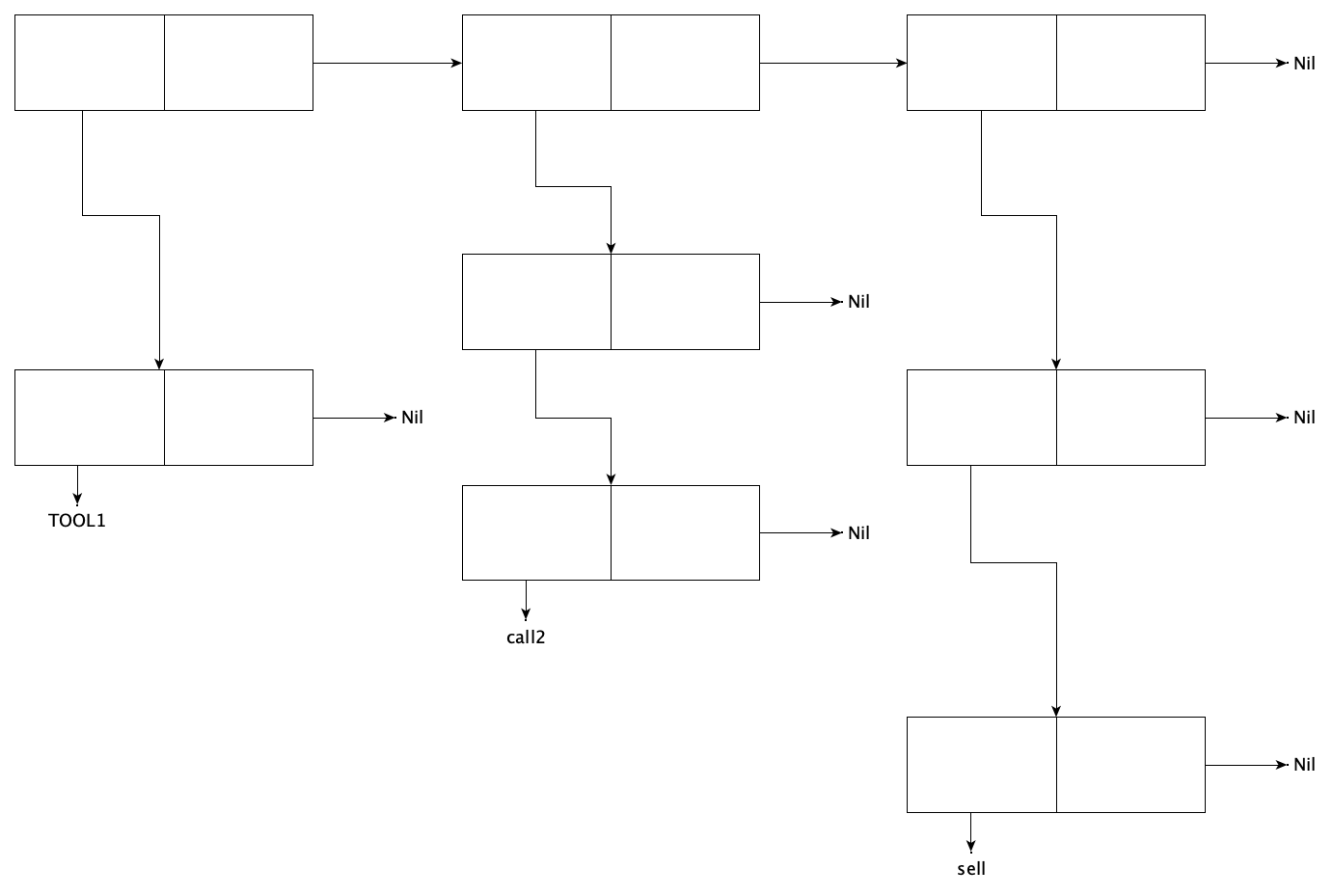
' ((TOOL) (call))



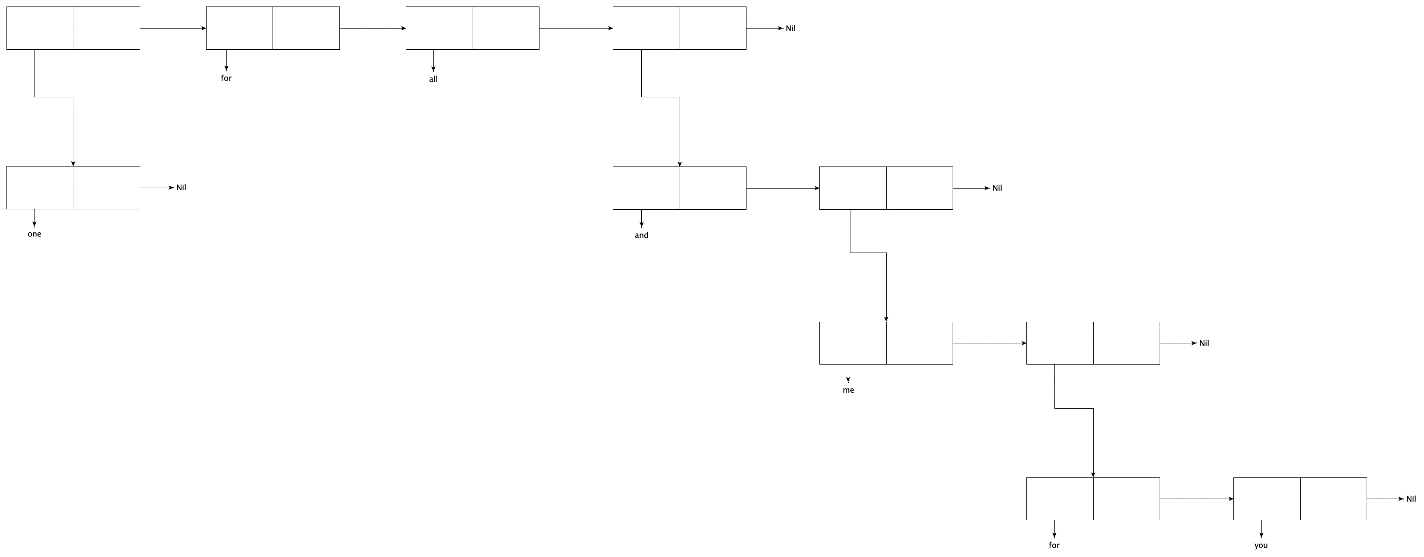
' ((open1) (close2) (halph3))



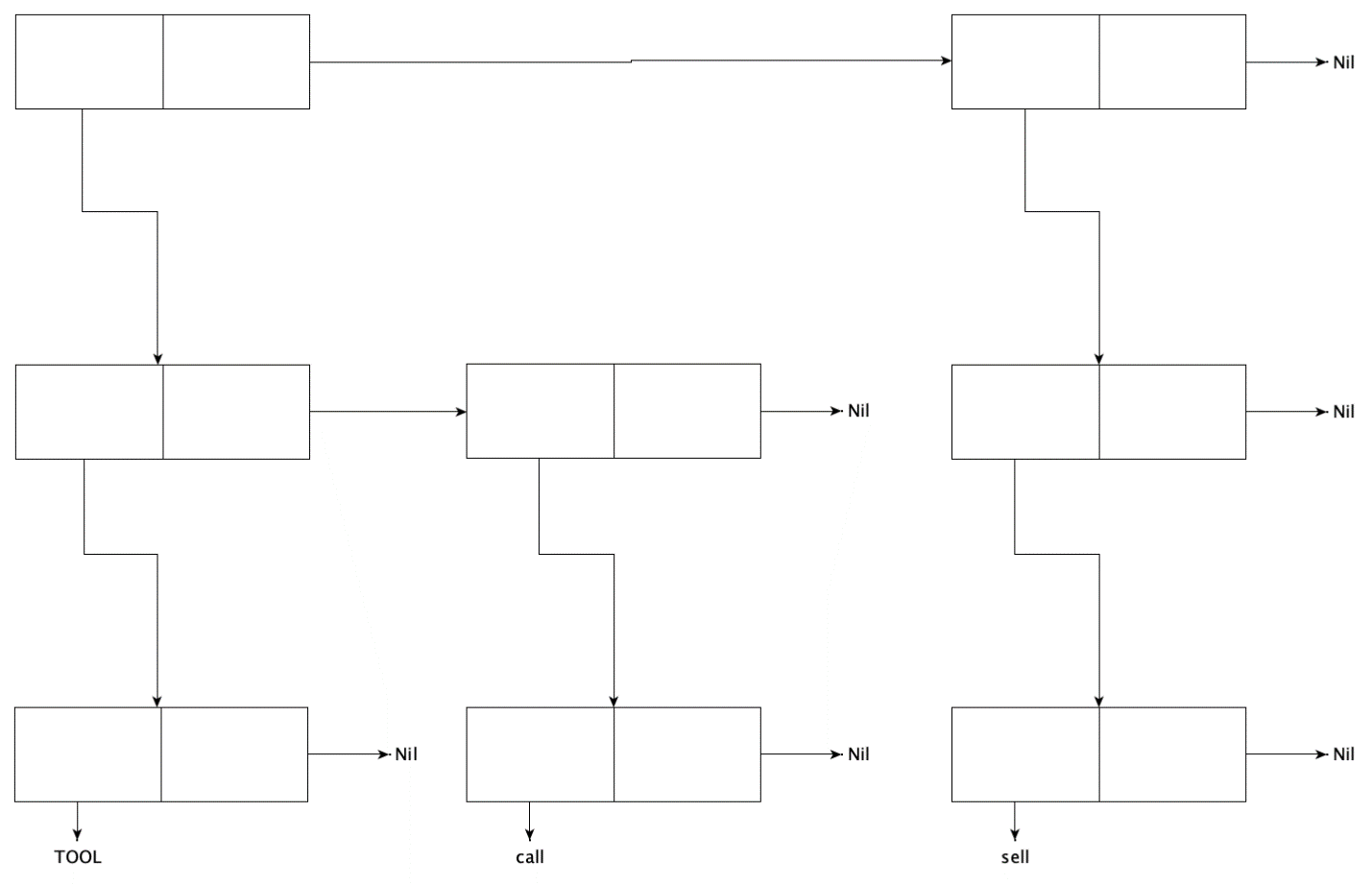
' (TOOL1)(call2)) ((sell)))



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



' (((TOOL) (call)) ((sell)))



## Задание №2

2. Используя только функции CAR и CDR, написать выражения,

возвращающие

1) второй - (CAR (CDR '(a b c d))

2) третий - (CAR (CDR (CDR'(a b c))))

3) четвертый элементы заданного списка. - (CAR (CDR (CDR (CDR '(a b c d)))))

## Ответы на вопросы

1. **Что такое список?** Список - это структура данных. Может быть пустой и непустой. Если непустой, то состоит из двух элементов: первый(голова) - любой структуры, а второй - список.
2. **Что такое базис?** Базис - это минимально необходимый набор конструкций с помощью которого можно запрограммировать.
3. **С помощью чего организованы списки в lisp?** С помощью бинарных узлов/списковых ячеек, содержащих пару указателей (на голову и хвост)
4. **В чем принципиальное отличие lisp от других языков?** Lisp - язык символьной обработки.
5. **Как воспринимается символ '?** Т.к. и программа и данные представлены списками, то их нужно как-то различать. Для этого была создана функция quote, а ' - ее сокращенное обозначение. quote - функция, блокирующая вычисление.
6. **Где в памяти находятся списки в lisp?** Списки динамические. В куче.