

*Государственное образовательное учреждение высшего  
профессионального образования*

**«Московский государственный технический  
университет имени Н.Э. Баумана»  
(МГТУ им. Н.Э. Баумана)**

ЛАБОРАТОРНАЯ РАБОТА №1

ПО КУРСУ «ФУНКЦИОНАЛЬНОЕ И ЛОГИЧЕСКОЕ  
ПРОГРАММИРОВАНИЕ»

## **Списки в Lisp. Использование стандартных функций.**

Выполнил: Тимонин А.С., гр. ИУ7-62Б

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

*2020 г.*

# Оглавление

Введение	2
Задание 1	3
Задание 2	4

# Введение

**Цель работы:** приобретение навыков работы со списками и стандартными функциями в Lisp.

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

**Базис Lisp** образуют: атомы, структуры, базовые функции, базовые функционалы.

Вся информация в Lisp представляется в виде символьных выражений. **Символьное выражение** - это атом или точечная пара.

**Атомами являются:** символы – набор литер, специальные символы – T, Nil, самоопределимые атомы – числа, строки.

Более сложные данные в Lisp выстраиваются с помощью **бинарных узлов**, содержащих пару указателей. Каждый бинарный узел соответствует минимальному блоку памяти, выделяемому системой при организации и обработке структур данных.

**Точечная пара** – структура данных, состоящая из двух символьных выражений, разделенных точкой.

**Список** – это структура данных. Может быть пустой и непустой. Если непустой, то состоит из двух элементов: первый - любой формы, а второй - список.

В памяти список представляется бинарным узлом, состоящим из двух указателей: `car` – указатель на первый элемент, `cdr` – указатель на оставшуюся часть.

Синтаксически любая структура (точечная пара или список) в языке Lisp заключается в круглые скобки. Точечная пара – (A.B). Пустой список можно задать пустыми скобками () или специальным символом `nil`. Непустой список можно задать через точечную пару (A.(B.())) (в этом случае происходит дублирование разделителей) или как последовательность атомов, разделенных пробелами (A B C).

# Задание 1

Изобразить следующие списки, используя списковые ячейки:

1. '(open close halph)
2. '((open1)(close2)(halph3))
3. '((one) for all (and(me(for you))))
4. '((TOOL)(call))
5. '((TOOL1)((call2)(sell)))
6. '(((TOOL)(call))((sell)))

## Задание 2

Используя только функции CAR и CDR, написать выражения, возвращающие:

1. второй
2. третий
3. четвертый элементы заданного списка

Листинг 1: Нахождение 2, 3, 4 элемента в списках

```
1 (car(cdr'(1 2 3 4 5))) ; 2  
2 (car(cdr(cdr'(1 2 3 4 5)))) ; 3  
3 (car(cdr(cdr(cdr'(1 2 3 4 5))))) ; 4
```