



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 № 5 по дисциплине «Функциональные и логические языки программирования»**

Тема Лисп. Представление списков.

Студент Одинцов Е.В.

Группа ИУ7-53БВ

Преподаватели Строганов Ю.В.

Москва, 2024

# Содержание

<b>ВВЕДЕНИЕ</b>	<b>4</b>
<b>1 Аналитическая часть</b>	<b>5</b>
<b>2 Технологическая часть</b>	<b>6</b>
2.1 Метод с использованием <code>cons</code>	6
2.2 Метод с использованием <code>list</code>	6
2.3 Метод с использованием <code>quote</code>	6
<b>ЗАКЛЮЧЕНИЕ</b>	<b>7</b>
<b>СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ</b>	<b>8</b>

# ВВЕДЕНИЕ

В данном отчете рассматривается разработка программы на языке программирования Common Lisp, в которой реализованы три способа создания вложенного списка

```
(1 2 (3 (((4))) 5) 6 (7 (8 9)) 0)
```

с использованием функций `cons`, `list` и `quote`. Целью является демонстрация возможностей создания сложных структур данных с использованием различных методов и функций языка.

## 1 Аналитическая часть

Для создания вложенного списка применяются три метода:

- 1) **С использованием функции** `cons` — данный метод позволяет пошагово построить список, добавляя элементы по одному. Это наиболее гибкий, но трудоемкий способ создания списка, так как требуется явное указание всех вложений.
- 2) **С использованием функции** `list` — позволяет создать список быстрее и проще, используя встроенную поддержку списков в Common Lisp. Этот метод более читабелен и удобен для создания сложных структур.
- 3) **С использованием** `quote` — позволяет создать литеральный список, что делает его самым коротким и быстрым способом для простых случаев, когда список известен заранее и не требует вычислений.

## 2 Технологическая часть

В данном разделе приведены реализации трех методов создания вложенного списка. Все функции написаны на языке Common Lisp.

### 2.1 Метод с использованием cons

Листинг 2.1 — Метод cons-list

```
(defun cons-list ()
  (cons 1
    (cons 2
      (cons (cons 3
        (cons (cons (cons (cons 4 nil) nil) nil)
          (cons 5 nil)))
        (cons 6
          (cons (cons 7
            (cons (cons 8
              (cons 9 nil))
              nil))
            (cons 0 nil))))))))))
```

### 2.2 Метод с использованием list

Листинг 2.2 — Метод list-method

```
(defun list-method ()
  (list 1 2
    (list 3
      (list (list (list 4)))
      5)
    6
    (list 7 (list 8 9))
    0))
```

### 2.3 Метод с использованием quote

Листинг 2.3 — Метод quote-method

```
(defun quote-method ()
  '(1 2 (3 (((4))) 5) 6 (7 (8 9)) 0))
```

# ЗАКЛЮЧЕНИЕ

В ходе выполнения работы были реализованы три метода создания вложенного списка на языке Common Lisp. Каждый из методов имеет свои преимущества и недостатки, что позволяет выбирать оптимальный подход в зависимости от задачи. Использование `cons` дает полную гибкость, `list` — упрощает создание списков, а `quote` — позволяет задавать структуры буквально.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Graham, P. (1995). *ANSI Common Lisp*. Prentice Hall.
2. Компилятор Steel Bank Common Lisp. Доступно на: <https://www.sbcl.org/>. [Дата обращения: октябрь 2024].
3. Библиотека Quicklisp beta. Доступно на: <https://www.quicklisp.org/beta/>. [Дата обращения: октябрь 2024].