



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 по лабораторной работе №3 по курсу «Функциональное и логическое программирование»

Тема Работа интерпретатора Lisp

Студент Прянишников А.Н.

Группа ИУ7-65Б

Оценка (баллы) \_\_\_\_\_

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

# Практическое задание

## Задание 1

Написать функцию, которая принимает целое число и возвращает первое четное число, не меньшее аргумента.

```
1 (defun first-even (val)
2   (if (evenp val) val (+ val 1))
3 )
```

## Задание 2

Написать функцию, которая принимает число и возвращает число того же знака, но с модулем на 1 больше модуля аргумента.

```
1 (defun next-val (val) (cond ((>= val 0) (+ 1 val)) ((< val 0) (- val 1))))
```

## Задание 3

Написать функцию, которая принимает два числа и возвращает список из этих чисел, расположенный по возрастанию.

```
1 (defun inc-list (a b)
2   (if (> a b) (list b a) (list a b))
3 )
```

## Задание 4

Написать функцию, которая принимает три числа и возвращает Т только тогда, когда первое число расположено между вторым и третьим.

```
1 (defun beetween (a b c)
2   (if (or
3       (and (> a b) (< a c))
4       (and (> a c) (< a b)))
5       T Nil))
6
7 (defun beetween (a b c)
8   (or
9     (and (> a b) (< a c))
10    (and (> a c) (< a b))))
```

## Задание 5

Каков результат вычисления следующих выражений?

```
1 #(
2   (and 'fee 'fie 'foe) ; FOE
3   (or 'fee 'fie 'foe) ; FEE
4   (and (equal 'abc 'abc) 'yes) ; YES
5   (or nil 'fie 'foe) ; FIE
6   (and nil 'fie 'foe) ; NIL
7   (or (equal 'abc 'abc) 'yes) ; T
8 )
```

## Задание 6

Написать предикат, который принимает два числа-аргумента и возвращает Т, если первое число не меньше второго.

```
1 (defun comp(a b) (>= a b))
```

## Задание 7

Какой из следующих двух вариантов предиката ошибочен и почему?

```
1 ; 1. Check on number
2 ; 2. Check on sign
3 (defun check (x)
4   (and (numberp x) (plusp x)))
```

## Задание 8

Решить задачу 4, используя для ее решения конструкции IF, COND, AND/OR.

```
1 (defun beetween-cond (a b c)
2   (cond ((or
3     (and (< a b) (> a c))
4     (and (> a b) (< a c))) T)
5     (T Nil)))
6
7 (defun beetween-andor (a b c)
8   (or
9     (and (< a b) (> a c))
10    (and (> a b) (< a c))))
```

## Задание 9

Переписать функцию how-alike, приведенную в лекции и использующую COND, используя только конструкции IF, AND/OR.

```
1 (defun how_alike_new (x y)
2 (if (or (= x y) (equal x y)) 'the_same
3     (if (and (oddp x) (oddp y)) 'both_odd
4         (if (and (evenp x) (evenp y))
5             'both_even 'difference))))
6
7 ; Only If
8 (defun how_alike_if (x y)
9 (if (if (= x y) (equal x y)) 'the_same
10     (if (if (oddp x) (oddp y)) 'both_odd
11         (if (if (evenp x) (evenp y))
12             'both_even 'difference))))
13
14
15 ; Only OR
16 (defun how_alike_and_or (x y)
17 (or (and (or (= x y) (equal x y)) 'the_same)
18     (and (and (oddp x) (oddp y)) 'both_odd)
19     (and (and (evenp x) (evenp y)) 'both_even))
20 'difference))
```

# Теоретические вопросы

## Базис языка Lisp

Базис - минимальный набор конструкций языка и структур данных, который позволяет решить любую задачу.

Базис в Lisp образуют:

- атомы;
- структуры;
- базовые функции;
- функционалы.

## Классификация функций языка Lisp

Функции в языке Lisp:

- Базовые/чистые функции – фиксированное кол-во аргументов, для определенного набора аргументов один фиксированный результат.
- Формы - функции, которые принимают произвольное количество аргументов или по разному обрабатывают аргументы.
- Функционалы (высшего порядка) - в качестве аргумента принимают функцию или возвращают функцию.

## Способы создания функций

С помощью макро определения `defun` или с использованием Лямбда-нотации (функция без имени).

# Работа функций `and`, `or`, `if`, `cond`

## Функция `and`

Синтаксис: `(and expression-1 expression-2 ... expression-n)`

Функция возвращает первое `expression`, результат вычисления которого `= Nil`. Если все не `Nil`, то возвращается результат вычисления последнего выражения.

## Функция `or`

Синтаксис: `(or expression-1 expression-2 ... expression-n)`

Функция возвращает первое `expression`, результат вычисления которого не `Nil`. Если все `Nil`, то возвращается `Nil`.

## Функция `if`

Синтаксис: `(if condition t-expression f-expression)`

Если вычисленный предикат не `Nil`, то выполняется `t-expression`, иначе - `f-expression`.

## Функция `cond`

По порядку вычисляются и проверяются на равенство с `Nil` предикаты. Для первого предиката, который не равен `Nil`, вычисляется находящееся с ним в списке выражение и возвращается его значение. Если все предикаты вернут `Nil`, то и `cond` вернет `Nil`.