



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»

НАПРАВЛЕНИЕ ПОДГОТОВКИ _____ «09.03.04 Программная инженерия»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №2

Название: _____ Определение функций пользователя

Дисциплина: _____ Функциональное и логическое программирование

Студент	ИУ7-66Б	_____	Т. А. Казаева
	Группа	Подпись, дата	И. О. Фамилия

Преподаватель	_____	Н. Б. Толпинская
	Подпись, дата	И. О. Фамилия

Москва, 2022 г.

1. ТЕОРЕТИЧЕСКИЕ ВОПРОСЫ

1. *Базис Lisp.*

- а) атомы и структуры (представляющиеся бинарными узлами);
- б) несколько базовых функций и функционалов: встроенные — примитивные функции (`atom`, `eq`, `cons`, `car`, `cdr`); специальные функции и функционалы (`quote`, `cond`, `lambda`, `eval`, `apply`, `funcall`).

2. *Классификация функций.*

- а) чистые (математические) функции: имеют фиксированное количество аргументов и в качестве возврата единственное значение;
- б) рекурсивные функции;
- с) специальные функции (формы): имеют произвольное количество аргументов, либо эти аргументы обрабатываются не все одинаково;
- д) псевдофункции: функции, эффект которых виден на внешних устройствах;
- е) функции с вариантными значениями, из которых выбирается одно;
- ф) функции высших порядков (функционалы) используются для построения синтаксически-управляемых программ, в качестве одного из аргументов принимают описание функции.

3. *Способы создания функций*

Обычно функции определяются при помощи макроса `DEFUN`. В качестве имени может использоваться любой символ. Как правило, имена функций содержат только буквы, цифры и знак минус. Список параметров функции определяет переменные, которые будут использоваться для хранения аргументов, переданных при вызове функции. Тело `DEFUN` состоит из любого числа выражений Lisp.

4. *Функции Car и Cdr*

Функция `car` служит для вывода первого элемента списка, а `cdr` — для вывода всех элементов, кроме первого.

5. *Назначение и отличие в работе Cons и List*

Функция `cons` принимает два аргумента и создаёт бинарный узел, первая ячейка которого указывает на первый переданный аргумент, а вторая ячейка — на второй.

Функция `list` не имеет ограничений по количеству передаваемых ей аргументов. Данная функция создаёт список, элементами которого являются все переданные функции аргументы.

`cons` работает эффективнее `list`, `list` описан с использованием `cons`.

2. ПРАКТИЧЕСКИЕ ЗАДАНИЯ

1. Составить диаграмму вычисления следующих выражений:

- | | |
|--|--|
| 1) (<code>equal</code> 3 (<code>abs</code> - 3)) | 2) (<code>equal</code> (+ 1 2) 3) |
| 3) (<code>equal</code> (* 4 7) 21) | 4) (<code>equal</code> (* 2 3) (+ 7 2)) |
| 5) (<code>equal</code> (- 7 3) (* 3 2)) | 6) (<code>equal</code> (<code>abs</code> (- 2 4)) 3) |

Решение представлено на отдельном листе и приложено к отчету.

2. Написать функцию, вычисляющую гипотенузу прямоугольного треугольника по заданным катетам и составить диаграмму её вычисления.

```
1 (defun hypo (x y) (sqrt (+ (* x x) (* y y))))
```

Диаграмма вычисления представлена на отдельном листе и приложена к отчету.

3. Написать функцию, вычисляющую объем параллелепипеда по 3-м его сторонам, и составить диаграмму ее вычисления.

```
1 (defun volm (x y z) (* x y z))
```

Диаграмма вычисления представлена на отдельном листе и приложена к отчету.

4. Каковы результаты вычисления следующих выражений? (объяснить возможную ошибку и варианты ее устранения)

- | | |
|-----------------------------------|--|
| 1) (<code>list</code> 'a c) | 2) (<code>cons</code> 'a (b c)) |
| 3) (<code>cons</code> 'a '(b c)) | 4) (<code>caddy</code> (1 2 3 4 5)) |
| 5) (<code>cons</code> 'a 'b 'c) | 6) (<code>list</code> 'a (b c)) |
| 7) (<code>list</code> a '(b c)) | 8) (<code>list</code> (+ 1 '(<code>length</code> '(1 2 3)))) |

Решения:

- 1) Переменная `c` не определена. Возможное устранение ошибки: (`list` 'a 'c)
- 2) Переменные `b` `c` не определены. Возможное устранение ошибки: (`cons` 'a '(b c))
- 3) ((A B C))

- 4) Функция `caddy` не определена. Возможное устранение ошибки: `(caddr '(1 2 3 4 5))`
- 5) Функция `cons` принимает два параметра. Возможное устранение ошибки: `(cons 'a '(b c))` ИЛИ `(cons 'a (list 'b 'c))`
- 6) Переменные `b`, `c` не определены. Возможное устранение ошибки: `(list 'a '(b c))`
- 7) `(A (B C))`
- 8) Поскольку введен запрет на вычисление `(quote)`, конструкция `length '(1 2 3))` будет интерпретирована как данные, а не как функция. Возможное устранение ошибки: `(list (+ 1 (length '(1 2 3))))`

5. Написать функцию `longer-then` от двух списков-аргументов, которая возвращает `T`, если первый аргумент имеет большую длину.

```
1 (defun longer-then(f s)(> (length f)(length s)))
```

6. Каковы результаты вычисления следующих выражений?

- | | |
|--|---|
| 1) <code>(cons 3 (list 5 6))</code> | 2) <code>(list 3 'from 9 'lives (- 9 3))</code> |
| 3) <code>(+ (length for 2 too)) (car '(21 22 23))</code> | 4) <code>(cdr '(cons is short for ans))</code> |
| 5) <code>(car (list one two))</code> | 6) <code>(cons 3 '(list 5 6))</code> |
| 7) <code>(car (list 'one 'two))</code> | |

Решения:

- | | |
|---|------------------------------------|
| 1) <code>(3 5 6)</code> | 2) <code>(3 from 9 lives 6)</code> |
| 3) Переменная <code>FOR</code> не определена. | 4) <code>(is short for ans)</code> |
| 5) Переменная <code>ONE</code> не определена. | 6) <code>(3 list 5 6)</code> |
| 7) <code>one</code> | |

7. Дана функция `(defun mystery (x)(list (second x)(first x)))`. Какие результаты вычисления следующих выражений?

- | | |
|-------------------------------------|--|
| 1) <code>(mystery (one two))</code> | 2) <code>(mystery (last one two))</code> |
| 3) <code>(mystery free)</code> | 4) <code>(mystery one 'two)</code> |

Решения:

- 1) Переменная ONE не определена. 2) Переменная ONE не определена.
3) Переменная FREE не определена. 4) Переменная ONE не определена.

8. Написать функцию, которая переводит температуру в системе Фаренгейта температуру по Цельсию (`defun f-to-c (temp)...`).

```
1 (defun f-to-c (temp) (* (/ 5 9) (- temp 32.0)))
```

Как бы назывался роман Р.Брэдли "451 по Фаренгейту" в системе по Цельсию?

Ответ: 232.77779 по Цельсию.

9. Что получится при вычисления каждого из выражений?

- 1) (list 'cons t NIL)
 - 2) (eval (eval (list 'cons t NIL)))
 - 3) (eval (list 'cons t NIL))
 - 4) (apply #cons "(t NIL)")
 - 5) (list 'eval NIL)
 - 6) (eval NIL)
 - 7) (eval (list 'eval NIL))

Решения:

- | | |
|-------------------------|---|
| 1) (cons t NIL) | 2) Переменная t не определена. |
| 3) t | 4) Неподдерживаемый синтаксис: #cons |
| 5) (eval NIL) | 6) NIL |
| 7) NIL | |

10. Дополнительно:

а) Написать функцию, вычисляющую катет по заданной гипотенузе и другому катету прямоугольного треугольника, и составить диаграмму ее вычисления.

```
1 (defun get-side (hypo side) (sqrt (- (* hypo hypo) (* side side))))
```

Диаграмма вычисления представлена на отдельном листе и приложена к отчету.

b) Написать функцию, вычисляющую площадь трапеции по ее основаниям и высоте, и составить диаграмму ее вычисления.

```
1 (defun trapez-area (a b h) (* (/ (+ a b) 2) h))
```

Диаграмма вычисления представлена на отдельном листе и приложена к отчету.