



Министерство образования и науки 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

ДИСЦИПЛИНА Функциональное и логическое программирование

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №7

Студент Зыкин Данила Андреевич

Группа ИУ7-63

Москва, 2019 г.

Задание 6(3)

```
(defun f(x y) (>= x y))
```

Задание 7(3)

Ошибочный 2ой, поскольку and заканчивает вычисление на 1ом nil

В 1ом варианте если введём строку то вычисление функции закончится and закончится на numberp.

Задание 8(3)

```
(defun f(a b c) (if (>= a b) (if (<= a c) t)))
```

```
(defun f(a b c) (cond ((>= a b) (<= a c))))
```

```
(defun f(a b c) (and (>= a b) (<= a c)))
```

Задание 9(3)

```
(defun how – alike(x y) (if (or (= x y) (equal x y)) '(the same)
    (if (and (oddp x) (oddp y)) '(both odd)
        (if (and (evenp x) (evenp y)) '(both even)
            '(difference)))))
```

Задание 1(4)

Дано

```
(setf lst1 '(a b))
```

```
(setf lst2 '(c d))
```

Выражение	Результат
(cons lst1 lst2)	((a b) c d)
(list lst1 lst2)	((a b) (c d))
(append lst1 lst2)	(a b c d)

Задание 2(4)

Выражение	Результат
(reverse())	nil
(last ())	nil
(reverse '(a))	(a)
(last '(a))	(a)

<code>(reverse '((a b c)))</code>	<code>(a b c)</code>
<code>(last '((a b c)))</code>	<code>(a b c)</code>

Задание 3(4)

```
(defun f(x) (car (reverse x)))
```

```
(defun f(x) (if (equal (cdr x) nil) (car x) (f (cdr x))))
```

Задание 4(4)

```
(defun f(x) (reverse (cdr (reverse x))))
```

Задание 5(4)

```
(defun roll() (cons (+ (random 6) 1) (+ (random 6) 1)))

(defun sum(x) (+ (car x) (cdr x)))

(defun reroll(x) (if (or (and (eq (car x) 1) (eq (cdr x) 1))
                        (and (eq (car x) 6) (eq (cdr x) 6))) t))

(defun won(x) (if (or (eq (sum x) 7) (eq (sum x) 11)) t))

(defun play() (setq p (roll))
              (if (won p) t
                  (if (reroll p) (play) (sum p))))

(defun start()
  (setq p1 (play))
  (setq p2 (play))
  (if (not (numberp p1)) (print "1st player total win")
      (if (not (numberp p2)) (print "2nd player total win")
          (if(= p1 p2) (format t "draw ~D~%" p1)
              (if(< p1 p2) (format t "2nd player won ~D//~D~%" p2 p1)
                          (format t "1st player won ~D//~D~%" p1 p2)))))))
```