

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«Санкт-Петербургский государственный технологический институт
(технический университет)»

Факультет: Информационных технологий и управления

Кафедра: Систем автоматизированного проектирования и управления

Уровень подготовки: Бакалавр

Учебная дисциплина: Математическая логика и теория алгоритмов

ПРАКТИЧЕСКАЯ РАБОТА № 6

тема: ЗНАКОМСТВО С LISP

Преподаватель _____ Плонский В.Ю.

Студент _____ Гусев А.А. 494 группа

Санкт-Петербург

2021

Цель: познакомиться с основами функционального программирования и базовым синтаксисом языка Lisp; изучить систему программирования «DrRacket»; исследовать основные механизмы абстракции процедур и данных.

Рассматриваются: вычисление выражений и комбинаций; подстановочная модель вычислений; составные процедуры; рекурсивные и итеративные процессы; процедуры высших порядков; λ -процедуры.

Ход работы

Задание 1

Объявлены следующие переменные:

```
(define X 2) (define Y 3)
(let ((a (+ 1 (* X Y)))
      (b (- 1 Y)))
  (+ (* X a a)
     (* Y b)
     (* a b)))
```

Необходимо переписать данный код, используя λ -процедуры.

```
((lambda (f g a b x y) (f (g x a a) (g y b) (g a b))) + * ((lambda (f
one g x y) (f one (g x y))) + 1 * X Y) ((lambda (f one y) (f one y)) - 1
Y) X Y)
```

Задание 2

Необходимо нарисовать стрелочные диаграммы и построить с помощью оператора cons следующие выражения: ((A B) C), ((A B) (D C)), ((A B) (D (C E))).

```
(cons (cons 'A 'B) 'C)
(cons (cons 'A 'B) (cons (cons 'C 'D) null))
(cons (cons 'A 'B) (cons (cons 'D (cons (cons 'C 'E) null))
null))
```

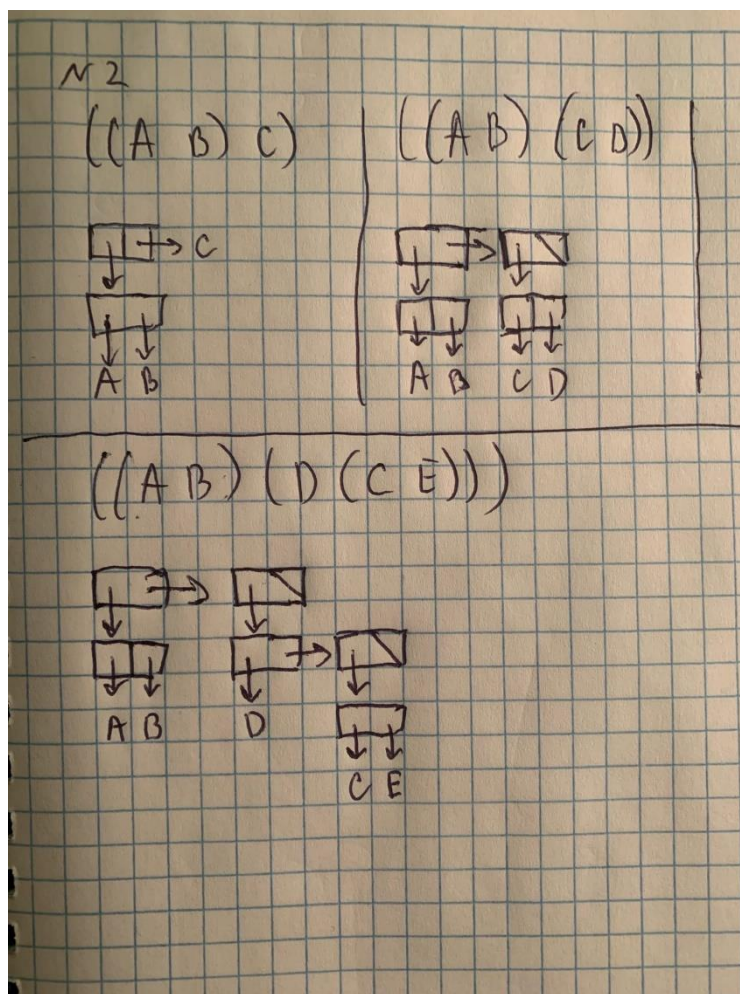


Рисунок 1 - Стрелочная диаграмма к заданию 2

Задание 3

Задан список, объявленный следующим образом:

`(LENGTH (cons (list 1 2) (list 3 4)))`, где `LENGTH` - функция, подсчитывающая число элементов в списке.

Необходимо нарисовать стрелочную диаграмму для данного списка и объяснить полученный результат.

Результат:

Запись `LENGTH (cons (list 1 2) (list 3 4))` эквивалентна записи `(LENGTH (cons (cons 1 (cons 2 null)) (cons 3 (cons 4 null))))`. Список, объявленный таким образом имеет вид `((1 2) 3 4)`, следовательно имеет результат 3 элемента, т.к. `(1 2)` является вложенным списком, который в свою очередь является первым элементом общего списка.

Задание 6

Определите процедуру, которая принимает в качестве аргументов три числа и возвращает сумму квадратов двух больших из них.

```
(define (func3 a b c)
  (if (> a b) (if (> b c) (+ (* a a) (* b b)) (+ (* a a) (* c
c)))
  (if(> a c) (+ (* a a) (* b b)) (+ (* b b) (* c c)))))
```

Задание 7

Определите процедуру, которая принимает в качестве аргументов два числа и возвращает сумму первого и абсолютное значение второго.

Замечание: отдельная процедура вычисления модуля числа (abs), а также встроенная процедура вычисления модуля не должна использоваться.

```
(define (func4 a b)
  (if (> 0 b) (+ a (- b)) (+ a b)))
```

Задание 8

Создайте процедуру для расчета функции Аккермана.

```
(define (func5 m n)
  (cond
    [(= m 0) (+ n 1)]
    [(= n 0) (func5 (- m 1) 1)]
    [else (func5 (- m 1) (func5 m (- n 1)))]))
```

Таблица 1 - Результат расчета функции Аккермана.

m	n	Время работы	Значение
0	0	< 1 с.	1
	1	< 1 с.	2
	2	< 1 с.	3
	3	< 1 с.	4
	4	< 1 с.	5
	5	< 1 с.	6
	6	< 1 с.	7
	7	< 1 с.	8
	8	< 1 с.	9
	9	< 1 с.	10
	n		n + 1
1	0	< 1 с.	2
	1	< 1 с.	3
	2	< 1 с.	4

Продолжение таблицы 1

m	n	Время работы	Значение
	3	< 1 с.	5
	4	< 1 с.	6
	5	< 1 с.	7
2	0	< 1 с.	3
	1	< 1 с.	5
	2	< 1 с.	7
	3	< 1 с.	9
	4	< 1 с.	11
	5	< 1 с.	13
3	0	< 1 с.	5
	1	1 с.	13
	2	1 с.	29
	3	1 с.	61
	4	1 с.	125
	5	1 с.	253
4	0	1 с.	13
	1	> 1 мин.	65533
	2	>> 1 мин.	$2^{65536} - 3$
5	0	> 1 мин.	65533

Задание 9

Напишите процедуру, вычисляющую элементы треугольника Паскаля с помощью рекурсивного процесса.

```
(define (factorial n)
  (if (= n 0) 1 (* n (factorial (- n 1)))))

(define (binom n m)
  (/ (factorial n) (* (factorial m) (factorial (- n m)))))

(define (pascal n k)
  (cond
    [(= k n) 1]
    [else (display (binom n k)) (pascal n (+ k 1))]))
```

Задание 10

Синус угла (заданного в радианах) можно вычислить, если воспользоваться приближением $\sin x \approx x$ при малых x и употребить тригонометрическое тождество для уменьшения значения аргумента \sin .

$$\sin(x) = 3\sin\left(\frac{x}{3}\right) - 4\sin^3\left(\frac{x}{3}\right)$$

```
(define (cube x) (* x x x))
(define (func x) (- (* 3 x) (* 4 (cube x))))
(define (sin10 a)
  (if (< (abs a) 0.1) (println "угол должен быть больше 0.1
радиан")
      (func (sin (/ a 3.0)))))
```

Вывод:

```
(sin10 0.01)
(sin10 3.142)
"угол должен быть больше 0.1 радиан"
-0.0004073463989415771
```

Выводы

При выполнении данной практической работы были получены базовые знания функционального программирования на примере языка LISP, изучен синтаксис языка и принципы работы в среде разработки «DrRacket». Также рассмотрены механизмы абстракции данных.