МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ И ПРОГРАММИРОВАНИЯ

ЗАЩИЩЕНА С ОЦЕНКОЙ

РУКОВОДИТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| старший преподаватель |  |  |  | Степанов П.А. |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| Лабораторная работа №2 |
| «Лямбда - выражения и функции в языке LISP» |
| по дисциплине: Технологии разработки серверных информационных систем |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4736 |  |  |  | Хмелев Н.А. |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2019**Цель работы** - ознакомление с лямбда - выражениями и функциями языка Lisp.

**Указания**. В работе не допускается использовать следующие функции:

операторы циклов; функции прямого доступа к элементам список, такие как nth,

elt, aref, высокоуровневые операторы обработки списков, такие как append,

reverse, nconc, функционалы, все виды оператора set, оператор prog.

Пример. Разработать функцию, разворачивающую список любой вложенности в

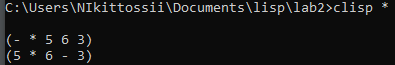
линейный в обратном порядке

Функция plainlist осуществляет первичную обработку списка, приводя его в

плоский вид. Res – промежуточный результат, lst –список**Вариант:**

16. Разработать функцию, выполняющую преобразование, обратное

предыдущему.Результат:



Выполнение:

(DEFUN A(LST)

    (B (CAR LST) (CDR LST))

)

(DEFUN B(ELEM LST)

    (COND

        ((EQL ELEM '+) (cons

                            (B (CAR (UNCDR LST)) (CDR (UNCDR LST)))

                            (CONS ELEM (cons (UNCAR LST) nil )))

                        )

        ((EQL ELEM '-)(cons

                            (B (CAR (UNCDR LST)) (CDR (UNCDR LST)))

                            (CONS ELEM (cons (UNCAR LST) nil )))

                        )

        ((EQL ELEM '\*)(cons

                            (B (CAR (UNCDR LST)) (CDR (UNCDR LST)))

                            (CONS ELEM (cons (UNCAR LST) nil )))

                        )

        ((EQL ELEM '/)(cons

                            (B (CAR (UNCDR LST)) (CDR (UNCDR LST)))

                            (CONS ELEM (cons (UNCAR LST) nil )))

                        )

        (T

            (CONS ELEM NIL)

        )

    )

)

(DEFUN UNCAR(LST)

    (COND

        ((NULL (CDR LST))

            (CAR LST)

        )

        (T

            (UNCAR (CDR LST))

        )

    )

)

(DEFUN UNCDR(LST)

    (COND

        ((NULL (CDR LST))

            nil

        )

        (T

            (CONS (CAR LST) (UNCDR ( CDR LST)))

        )

    )

)

(SETQ test '(- \* 5 6 3))

(print test)

(PRINT (A test))