Министерство науки и высшего образования РФ

Севастопольский государственный университет

Кафедра информатики и управления в технических системах

ОТЧЕТ

по лабораторной работе №6

Разработка и исследование файлового ввода-вывода на Лиспе

по дисциплине «Логическое программирование»

Выполнил:

Студент группы ИВТ/б 22-о

Черняев Н.Г.

Проверил:

Ткаченко К.С.

г. Севастополь 2019

**Цель работы**

Изучить основы программирования файлового ввода-вывода.

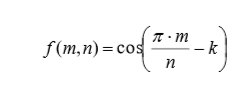
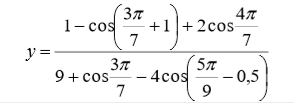
**Вариант №23,7,17,24**

Выполнить лабораторную работу № 2, исходные данные вводятся из файла, результат выводится в файл и на консоль.

Вычислить заданную зависимость y на основе функции f.

Функция f(x) должна быть определена с помощью defun.

Задание для варианта №23:



Вычислим y на калькуляторе:

y=0.2

Текст программы:

(print (format NIL "Чтение файла"))

(with-open-file (stream "F:\\lisp\\in23.txt" :direction :input)

(let ((a1 (parse-integer(read-line stream)))

(b1 (parse-integer(read-line stream)))

(c1 (parse-integer(read-line stream)))

(a2 (parse-integer(read-line stream)))

(b2 (parse-integer(read-line stream)))

(c2 (parse-integer(read-line stream)))

(a3 (parse-integer(read-line stream)))

(b3 (parse-integer(read-line stream)))

(c3 (parse-integer(read-line stream)))

(a4 (parse-integer(read-line stream)))

(b4 (parse-integer(read-line stream)))

(c4 (parse-integer(read-line stream))))

(defun f23(m n k) (cos (- (/ (\* pi m) n) k)))

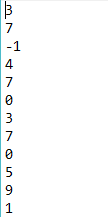
(defun y23() (/ (+ (- 1 (f23 a1 b1 c1)) (\* 2 (f23 a2 b2 c2))) (+ 9 (- (f23 a3 b3 c3) (\* 4 (f23 a4 b4 c4))))))

(setq y (y23))))

(with-open-file (stream "F:\\lisp\\out23.txt" :direction :output :if-exists :supersede)

(format stream "~$" y))

Данные содержащиеся в файле in23.txt



Вывод результата в файл out23.txt



Задание для варианта №7:

Вычислим y на калькуляторе для x=1:

y=0.99

Текст программы:

(print (format NIL "Чтение файла"))

(with-open-file (stream "F:/lisp/in23.txt" :direction :input)

(let ((x (parse-integer(read-line stream)))

(a1 (parse-integer(read-line stream)))

(b1 (parse-integer(read-line stream)))

(c1 (parse-integer(read-line stream)))

(a2 (parse-integer(read-line stream)))

(b2 (parse-integer(read-line stream)))

(c2 (parse-integer(read-line stream))))

(defun f7(a b c x)(+ (\* a (expt x 3)) (\* b x) c))

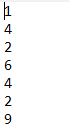
(defun y7 (x) (/ (f7 a1 b1 c1 x) (f7 a2 b2 c2 x)))

(setq y (y7 x))))

(with-open-file (stream "F:/lisp/out7.txt" :direction :output :if-exists :supersede)

(format stream "~$" y)(close stream))

Данные содержащиеся в файле in7.txt



Вывод результата в файл out7.txt



Задание для варианта №17:

Вычислим y на калькуляторе для x=1:

y=0.25

Текст программы:

(print (format NIL "Чтение файла"))

(with-open-file (stream "F:/lisp/in17.txt" :direction :input)

(let ((x (parse-integer(read-line stream)))

(a1 (parse-integer(read-line stream)))

(b1 (parse-integer(read-line stream)))

(c1 (parse-integer(read-line stream)))

(a2 (parse-integer(read-line stream)))

(b2 (parse-integer(read-line stream)))

(c2 (parse-integer(read-line stream))))

(defun f17(a b c x) (- (\* a (expt x 3)) (\* b (expt x 2)) c))

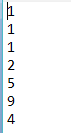
(defun y17 (x) (/ (f17 a1 b1 c1 x) (f17 a2 b2 c2 x)))

(setq y(y17 x))))

(with-open-file (stream "F:/lisp/out17.txt" :direction :output :if-exists :supersede)

(format stream "~$" y))

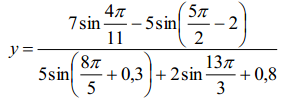
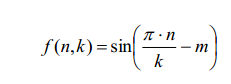
Данные содержащиеся в файле in17.txt



Вывод результата в файл out17.txt



Задание для варианта №24:

Вычислим y на калькуляторе:

y= −3.8

Текст программы:

(print (format NIL "Чтение файла"))

(with-open-file (stream "F:/lisp/in24.txt" :direction :input)

(let ((a1 (parse-integer(read-line stream)))

(b1 (parse-integer(read-line stream)))

(c1 (parse-integer(read-line stream)))

(a2 (parse-integer(read-line stream)))

(b2 (parse-integer(read-line stream)))

(c2 (parse-integer(read-line stream)))

(a3 (parse-integer(read-line stream)))

(b3 (parse-integer(read-line stream)))

(c3 (parse-integer(read-line stream)))

(a4 (parse-integer(read-line stream)))

(b4 (parse-integer(read-line stream)))

(c4 (parse-integer(read-line stream))))

(defun f24(n k m) (sin (- (/ (\* pi n) k) m)))

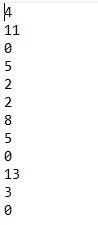
(defun y24() (/ (- (\* 7 (f24 a1 b1 c1)) (\* 5 (f24 a2 b2 c2))) (+ (\* 5 (f24 a3 b3 c3)) (\* 2 (f24 a4 b4 c4)) 0.8)))

(setq y (y24))))

(with-open-file (stream "F:/lisp/out24.txt" :direction :output :if-exists :supersede)

(format stream "~$" y))

Данные содержащиеся в файле in24.txt



Вывод результата в файл out24.txt



**Выводы**

В данной лабораторной работе были изучены основы программирования ввода-вывода на Лиспе. Результаты работы программы и прдеварительные вычисления на калькуляторе полностью совпали.