Raport Formal Languages & Translators

Missing Lab Recovery

Neagu Lorena

Group 30431

The program represents a lisp microinterpretor.

The program contains 2 files : lisp.l & lisp.y for the lisp microinterpretor.

The program is compiled executing the following commands on the program’s files:

-lex lisp.l

-yacc -d lisp.y

-gcc -o LISP lex.yy.c y.tab.c -ly –ll

The program is run by the following command:

./LISP + input

Example input : (CONS 1 '(2 3 )) will output => [1 2 3]

(CONS 2 '(9 10 11)) will output => [2 9 10 11]

(CAR '(1 1 3)) will output => 1

(CDR '(1 2 3)) will output => [2 3]

The program creates a lisp microcontroller by creating lisp commands fron scratch, like CONS, CAR, CDR, APPEND, having as arguments integers or lists (i\_form, l\_form).

The program contains the declarations and terminal and nonterminal definitions, followed by a set of rules and subroutines that define the lisp representative commands, described in lisp.y file.

The file lisp.l is like a lisp compiler, in which it is specified what type of values can be input, it can be numbers or lists. It also defines the lisp commands CONS, CAR, CDR, APPEND.

The result will be nicely printed to the screen.