Homework #4: CMPT-379

Distributed on Mon, Feb 9; Due on Mon, Feb 23

Anoop Sarkar – anoop@cs.sfu.ca

(1) **LR Parsing**: Implement the LR parsing algorithm (see Section 4.7 of the Dragon book). Your program should read in a text file containing a context-free grammar (the textual format of CFGs was described in a previous homework). In addition the LR parser should read in two text files, one containing the *action* table, the other containing the *goto* table. The output of the LR parser should be the parse tree corresponding to the input string.

For this homework, instead of starting with the full **`Decaf`** grammar, your LR parser should be tested with the example grammar shown in Example 4.33 of the Dragon book.

Your program should run as follows:

```
cat tokens | parser exprGrammar.txt action.txt goto.txt
```

For the following token input:

```
ID x
PLUS +
ID y
TIMES *
LPAREN (
ID z
RPAREN )
```

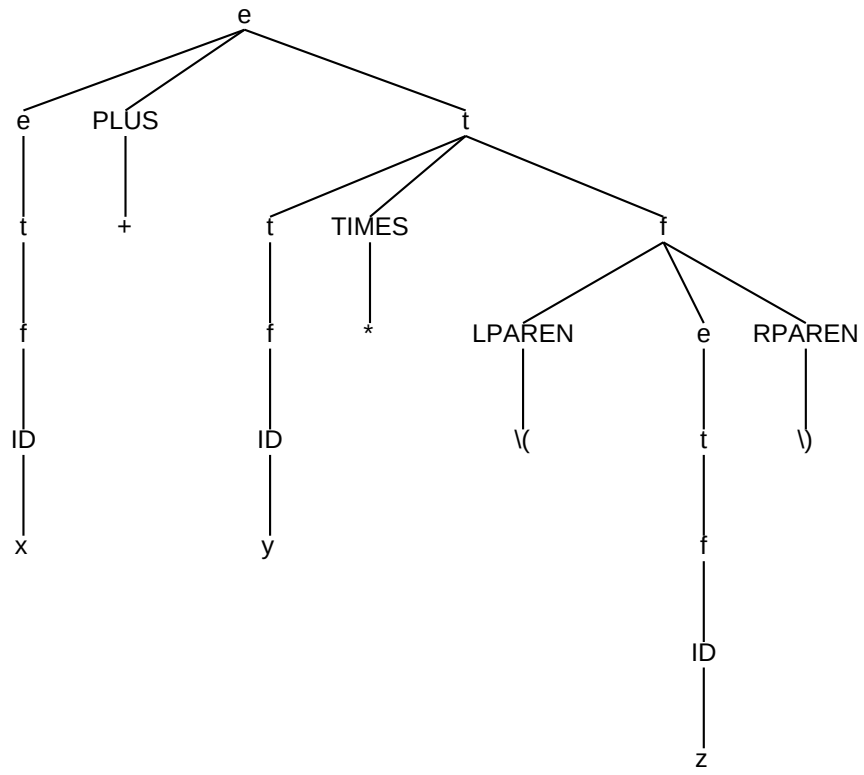
and given the expression grammar from Example 4.33 in the Dragon book, written in the usual text format below:

```
e e PLUS t
e t
t t TIMES f
t f
f LPAREN e RPAREN
f ID
```

the LR parser output should be the parse tree in the format shown below. Note the backslash preceding each instance of a literal parenthesis to avoid confusion with the parentheses used to denote the tree structure.

```
(e (e (t (f (ID x))))
   (PLUS +)
   (t (t (f (ID y)))
      (TIMES *)
      (f (LPAREN \()
         (e (t (f (ID z))))
         (RPAREN \)))))
```

For simplicity, your output parse tree can be printed out as one parse tree per line, rather than the indented form shown above. In a graphical view, the above parse tree will be as shown below:



Hints on code design:

- Your first step should be to implement a data structure for CFGs. You will need to read in files containing CFGs in the text format.
- When designing the data structure for CFGs you should pay attention to the future use of this data structure. In particular, consider the efficient implementation of the *closure* operation described in Figure 4.33 in the Dragon book.
- When designing the data structure for the *goto* table, notice that the *goto* table is identical to the definition of DFAs.
- When designing the data structure for the *action* table, notice that the *action* table is only a slight variation from the definition of DFAs. Assume a standard indexing scheme for the CFG rules for the *reduce* action when defining the text file for the *action* table.