4 Compiler Construction (tgg22)

(a) Show that the following grammar for the language of balanced parenthesis is ambiguous.

\[
S \rightarrow \epsilon \\
   | (S) \\
   | SS
\]

[5 marks]

(b) An arithmetic expression such as \( x \ast ((z \ast u) + y) \) can be represented without parenthesis in postfix notation as \( xz\ast u \ast y + \ast \). This representation is ideal for evaluation using a stack machine.

Below is a simple grammar for postfix expressions:

\[
S \rightarrow i \\
   | SS+ \\
   | SS*
\]

The terminal \( i \) represents the lexical class of identifiers. Here is a DFA for the LR(0) items of this grammar.

(i) Using the DFA, construct the SLR(1) ACTION and GOTO tables for this grammar. Explain your work. [6 marks]

(ii) Show a trace of a parsing of \( w = iii \ast i + \ast \). Justify every step. [9 marks]