Computer Science 405
Fall 2018
Programming Assignment 2
Due: Friday, November 2

The input to this next program consists of syntactically correct sequence taken from the set (TRUE, FALSE, XOR, AND, OR), i.e., a Boolean expression containing only literals TRUE and FALSE and the three indicated operators. The output to the program will be the number of ways in which the Boolean expression can be parenthesized to render the expression True. For example, there are 2 ways to parenthesize

TRUE AND FALSE XOR TRUE

such that it evaluates to True.

Behind this problem, characteristic of problems solvable by dynamic programming, there is a recursive equation that supports a dynamic programming approach.

Let $T[i,\ j]$ be the number of ways of parenthesizing the string $S[i,j]$ such that the expression between indices i and j evaluates to True, and $F[i,\ j]$ be the number of ways of parenthesizing the string $S[i\ ,j]$ such that the expression between indices i and j evaluates to False.

Here indices i and j are inclusive. Here are the relevant equations:

$$Total[i,j] = \ Total[i,k] + \ Total[k+1,j]$$

$$\begin{matrix} T[i,j] \\ Total(i,j) \ = \ T(i,j) \ + \ F(i,j) \end{matrix}$$

$$T[i,j] = \sum_{k=i}^{j-1} \begin{cases} T[i,k] \cdot T[k+1,j] & \text{if } S[k] = \texttt{AND} \\ Tot[i,k] \cdot Tot[k+1,j] - F[i,k] \cdot F[k+1,j] & \text{if } S[k] = \texttt{OR} \\ T[i,k] \cdot F[k+1,j] + F[i,k] \cdot T[k+1,j] & \text{if } S[k] = \texttt{XOR} \end{cases}$$

$$F[i,j] = \sum_{k=i}^{j-1} \begin{cases} F[i,k] \cdot F[k+1,j] & \text{if } S[k] = \texttt{OR} \\ Tot[i,k] \cdot Tot[k+1,j] - T[i,k] \cdot T[k+1,j] & \text{if } S[k] = \texttt{AND} \\ T[i,k] \cdot T[k+1,j] + F[i,k] \cdot F[k+1,j] & \text{if } S[k] = \texttt{XOR} \end{cases}$$

With these boundary conditions:

$$\begin{matrix} T[i,i] = 1 & \text{if } S[i] = \texttt{True} \\ T[i,i] = 0 & \text{if } S[i] = \texttt{False} \\ F[i,i] = 0 & \text{if } S[i] = \texttt{True} \\ F[i,i] = 1 & \text{if } S[i] = \texttt{False} \end{matrix}$$

Stipulations: (1) Input file names should be read from the command line; (2) upload your Java source program uncompressed, (3) your program will be tested using jGrasp, (4) there will one input sequence per test file.