

Compiladores — Folha laboratorial 4

DCC/FCUP

Outubro 2024

Análise sintática LR

Exercício 1

Considere a seguinte gramática (ambígua) para expressões aritméticas simples:

$$\begin{array}{lll} E \rightarrow E+E & E \rightarrow E * E & E \rightarrow (E) \\ E \rightarrow E-E & E \rightarrow E/E & E \rightarrow \text{num} \end{array}$$

- (a) Implemente uma calculadora para expressões desta gramática usando geradores de analisadores léxicos e sintáticos. Use o *Alex/Happy* com a linguagem Haskell ou o *Flex/Bison* com a linguagem C.

Resolva as ambiguidades usando diretivas de associatividade e precedência. Tenha o cuidado de verificar se a resolução respeita as convenções algébricas usuais. Exemplos:

1+2*3	deve calcular $1 + (2 \times 3)$	(prioridade de $*$ em relação a $+$)
1+2/3	deve calcular $1 + (2/3)$	(prioridade de $/$ em relação a $+$)
1-2-3	deve calcular $(1 - 2) - 3$	(associatividade à esquerda)
1/2/3	deve calcular $(1/2)/3$	(associatividade à esquerda)

O seu programa não necessita de construir a *árvore sintática abstrata*; pode efetuar os cálculos durante a análise sintática. O resultado deve ser calculado em vírgula flutuante de precisão simples.

- (b) Modifique a análise léxica, sintática e a função de avaliação para acrescentar algumas funções matemáticas à calculadora:

$$\begin{array}{l} E \rightarrow \dots \mid F(E) \\ F \rightarrow \text{sqrt} \mid \text{exp} \mid \text{log} \end{array}$$