

TD d'analyse syntaxique: préparation du TP2

L'objectif du TP est de programmer en PYTHON la calculette spécifiée au chapitre 2 du cours d'amphi. On réutilise ici l'analyseur lexical du TP1 en ajoutant les tokens **OPAR** et **CPAR** (et en conservant le token **END**). On rappelle que la calculette est spécifiée par la BNF ci-dessous

$$\begin{aligned}
 \text{input} \uparrow \ell &::= \epsilon & \ell &:= [] \\
 &| \text{input} \uparrow \ell_0 \text{ exp} \downarrow \ell_0 \uparrow n \text{ QUEST} & \ell &:= \ell_0 \oplus n \\
 \\
 \text{exp} \downarrow \ell \uparrow n &::= \text{INT} \uparrow n & & \\
 &| \text{VAR} \uparrow i & n &:= \ell[i] \\
 &| \text{exp} \downarrow \ell \uparrow n_1 \text{ PLUS } \text{exp} \downarrow \ell \uparrow n_2 & n &:= n_1 + n_2 \\
 &| \text{exp} \downarrow \ell \uparrow n_1 \text{ MINUS } \text{exp} \downarrow \ell \uparrow n_2 & n &:= n_1 - n_2 \\
 &| \text{exp} \downarrow \ell \uparrow n_1 \text{ MULT } \text{exp} \downarrow \ell \uparrow n_2 & n &:= n_1 \times n_2 \\
 &| \text{exp} \downarrow \ell \uparrow n_1 \text{ DIV } \text{exp} \downarrow \ell \uparrow n_2 & n &:= n_1 / n_2 \\
 &| \text{MINUS } \text{exp} \downarrow \ell \uparrow n_0 & n &:= -n_0 \\
 &| \text{OPAR } \text{exp} \downarrow \ell \uparrow n \text{ CPAR} & &
 \end{aligned}$$

cette BNF étant désambiguïsée par la table de priorités suivante

niveau 2 (priorité min)	associatif à gauche	PLUS binaire MINUS binaire
niveau 1	associatif à gauche	MULT binaire DIV binaire
niveau 0 (priorité max)		MINUS unaire

▷ **Question 1.** En suivant les principes du chapitre 5 du cours, donner une BNF non-ambiguë qui définit la même syntaxe et la même sémantique que celle du chapitre 2 du cours (quand on prend en compte les priorités). La BNF obtenue est-elle LL(1)? Sinon, mettre cette BNF sous forme LL(1). NB : on peut aussi utiliser une EBNF LL(1). ◁

▷ **Question 2** (A PREPARER EN TEMPS LIBRE AVANT LA SEANCE DE TP). Programmer l'analyseur syntaxique (en suivant les principes du chapitre 4). ◁