

Expressions et Fonctions Lex

I. Expressions régulières de LEX

Expression	Désigne	Exemple
c	tout caractère c qui n'est pas un opérateur	a
\c	caractère littéral c	*
"s"	chaîne littérale s	"**"
.	Tout caractère sauf fin de ligne	a.*b
^	début de ligne	^abc
\$	fin de ligne	abc\$
[s]	tout caractère appartenant à s	[abc]
[^s]	tout caractère n'appartenant pas à s	[^abc]
r*	zéro ou plusieurs r	a*
r+	un ou plusieurs r	a+
r?	zéro ou un r	a?
r{m,n}	entre m et n occurrences de r	a{1,5}
r ₁ r ₂	r ₁ puis r ₂	ab
r ₁ r ₂	r ₁ ou r ₂	a b
(r)	r	(a b)
r ₁ /r ₂	r ₁ quand suivi de r ₂	abc/123

II. Quelques fonctions et variables de LEX

- L'analyse lexicale est effectuée par une fonction **yylex()** qui est contenue dans le fichier `lex.yy.c`
- La fonction **yylex()** qui doit être appelée pour utiliser l'analyseur lexical :
 - analyse séquentiellement un fichier d'entrée
 - retourne 0 lorsqu'elle rencontre une fin de fichier
 - effectue des opérations spécifiées par le programme Lex, lorsqu'une unité lexicale est reconnue

- LEX fournit un ensemble de variables et de routines :
 - Variable **yyval** de LEX permet de communiquer des valeurs à YACC.
 - **yytext** = variable contenant la chaîne de caractères courante qui a été reconnue
 - **yylen** = longueur de la chaîne **yytext**
 - **yyless(k)** = fonction admettant un entier comme argument
 - ✓ supprime les (yylen-k) derniers caractères de **yytext**, dont la longueur devient alors k
 - ✓ recule le pointeur de lecture sur le fichier d'entrée de **yylen-k** positions, les caractères supprimés de **yytext** seront donc considérés pour la reconnaissance des prochaines unités lexicales
 - **yymore()** = fonction qui concatène la chaîne actuelle **yytext** avec celle qui a été reconnue avant
 - **yywrap()** = fonction appelée lorsque **yylex()** rencontre une fin de fichier
 - ✓ si **yywrap()** retourne `true`, alors **yylex()** retourne 0 pour indiquer une fin de fichier
 - ✓ si **yywrap()** retourne `false`, alors **yylex()** ignore la fin de fichier et continue son analyse
 - ✓ Par défaut, **yywrap()** retourne 1, mais on peut la redéfinir.
- L'entrée standard est notée **yyin** et la sortie standard est notée **yyout**

Exemple : Si on veut que :

- l'entrée standard soit le fichier `toto.e` qui contiendra le texte à analyser
- la sortie standard soit le fichier `toto.s` qui contiendra les sorties (messages ...) générés :

```
errno_t Err1=fopen_s(&yyin," toto.e","r");
errno_t Err2=fopen_s(&yyout,"toto.s","w");
...
...
...
fclose(yyin);
fclose(yyout);
```