

Exercice 01

- i. Écrire un lexeur flex qui compte (dans fichier) le nombre de :
 - a. caractères,
 - b. de mots,
 - c. des identificateurs et
 - d. de lignes;
- ii. Écrire un lexeur flex qui ajoute des numéros aux lignes d'un fichier (sauf les lignes blanches) ;
- iii. Écrire un lexeur flex qui n'affiche que les commentaires d'un programme. Ceux-ci sont compris entre {};
- iv. Écrire un lexeur flex qui remplace dans un texte le mot groupe par section si la ligne début par a, par classe si la ligne débute par b.

Exercice 02

Écrire une fonction d'analyse lexicale à l'aide de LEX. Les *tokens* reconnus seront :

- Les nombres décimaux (en notation scientifique) ;
- Les identifiants de variables ;
- Les opérateurs relationnels RELOP (< : PPQ, > : PGQ, <= : PPE, >= : PGE, <> : DIF) ;
- Les mots-clefs si, sinon et alors.

En utilisant les fonctions suivantes :

- RangerID : range l'identificateur dans la TS.
- RangerId() : // numéro

Cette fonction a pour but d'être utilisée dans l'analyse syntaxique.

Exercice 3 :

Écrire un lexeur flex capable de reconnaître les entités suivantes :

- Les mot-clefs : **begin**, **end** ;
- les identificateurs (séquences de chiffres et de lettres commençant par une lettre, et qui ne sont pas des mot-clefs) ;
- les opérateurs : +, -, * et **.

Les espaces, tabulations et retours à la ligne sont des séparateurs. Toute autre séquence de caractères qui ne forme pas une entité est considérée comme une erreur.

Le lexeur doit afficher à l'écran la liste des entités reconnues : "nombre 12", "identificateur abc", "erreur", etc.

Exercice 4:

Écrire un lexeur flex capable de compter les voyelles, consonnes et caractères de ponctuation d'un texte.

Exercice 5:

Écrire un lexeur FLEX qui insert le numéro de ligne à chaque ligne dans un fichier.

Exercice 6:

Écrire un lexeur FLEX qui reconnaît :

- les nombres Binaires.
- Les nombre du système octale.