

Série de TP n° 1

Exercice 1 :

L'analyse lexicale constitue la première phase d'un compilateur. La tâche principale d'un analyseur lexical est d'isoler, en fonction de modèles, les lexèmes et les unités lexicales correspondantes.

Ecrire un analyseur lexical qui reconnaît les lexèmes associés aux unités lexicales suivantes :

- les identificateurs, dénotés par l'expression régulière :

[a-zA-Z][a-zA-Z0-9]*

- les nombres dénotés par l'expression régulière :

(+|-)?[0-9]+(.[0-9]+)?((e|E)(+|-)?[0-9]+)?

- les opérateurs de relation : **< | <= | <> | == | > | >=**
- les mots clés : **var, begin, end, if, then, else, programm.**

Exercice 2:

La table des symboles est une structure de données contenant un enregistrement pour chaque lexème du texte d'entrée.

En utilisant les résultats de l'exercice 1, écrire un programme qui reçoit en entrée un nom d'un fichier, parcourt le texte du fichier et insère dans la table des symboles :

- tous les identificateurs,
- tous les mots clés, pour cela on prévoit une table des mots clés,
- tous les nombres,
- tous les opérateurs de relation
- les autres caractères seront traités comme des caractères inconnus.

La table des symboles aura la structure suivante :

```
{char* lexeme ; // le nom du lexème reconnu  
  attribut unilex ; // son type : id, mot_cle...
```

Avec attribut un type énuméré {id, mot_cle, operel, nb}

Exercice 3:

Reprendre les 2 exercices précédents avec le langage **FLEX**.