Traduction – Chapitre 2 : Analyse lexicale

# Analyse syntaxique

* Dans le schéma précédent, il s’agit de la première étape.
* C’est un programme, un automate à état, qui lit le texte source caractère par caractère.
  + Le seul programme qui lit le fichier source.
* Il est sensé reconnaitre les unités lexicales (TOKEN en anglais) : les mots du langage, et les présente à l’analyseur syntaxique.
* Exemples d’unités lexicales :
  + Constantes numériques
  + Chaines littérales, de deux types :
    - Identificateurs (de variables, de fonctions)
    - Mots-clés (if, for, then, …)
  + Caractères spéciaux
    - Simples ( + , - , \* , ( , … )
    - Doubles ( +=, <=, !=, …)
  + 🡺Terminaux de la grammaire
* Autres missions du lexeur :
  + Supprimer les espaces, les fins de lignes, les tabulations
  + Supprimer les commentaires /\* ---- \*/
  + Afficher les erreurs

# Erreurs lexicales possibles

* Un identificateur est mal formé
* Un caractère inconnu
* Commentaire non formé
* Constante ou entée trop grande

# Codage des unités lexicales

## Le lexique

* Le lexer code les unités lexicales, pour des raisons d’efficacité car il doit reconnaitre des mots.
* Utiliser des identificateurs lexicaux qui sont claires.
* Représentation économe :
  + Plus simple à mémoriser et à factoriser
* Le code des unités lexicales est représenté par des entiers.

Exemples :

* Unités simples :

|  |  |
| --- | --- |
| + | 1 |
| - | 2 |
| \* | 3 |
| / | 4 |
| := | 5 |

* Unités génériques
  + Identificateurs : 7
  + Constante entière : 8
* Mots clés
  + If : 9
  + For : 10
* Soit le code suivant
  + /\* un exemple \*/  
    x := y + 2 \* x
* Le code passera dans le Lexer
* Il en sortira une suite d’unité lexicale.
  + Sortie : (7, « x ») ; 5 ; (7, « y ») ; (8, 2), 3, (7, « x »)

Le résultat obtenu est donc une suite de Token

Le lexique est une table d’identificateur.

|  |  |
| --- | --- |
| 1 | « x » |
| 2 | « y » |
| 3 | « Factorielle » |
| … | … |
| N | … |

On obtient ensuite la sortie : (7, 1) ; 5 ; (7, 2) ; (8, 2), 3, (7, 1) => Construire un arbre abstrait

Dans (7, « x »), le 7 correspond au code donné par l’analyseur lexical. Le « x », on va le mettre dans un lexique et mettre à la place de la chaine « x », son rang dans le lexique.

Les caractères du clavier sont codés en code ascii.

Le lexique est à créer par nous-même …

Résumé :

* On attribue un numéro lexical[[1]](#footnote-1) à chaque identificateur.
* On retrouve, pour un numéro lexical la chaine de caractères ( = identificateur) associée

# Pourquoi coder les identificateurs ?

Supposons que nous avons : identificateur := expression

* Pour l’analyseur syntaxique, on doit trouver un identificateur en partie gauche (peu importe l’identificateur)
* Pour l’analyseur sémantique, on doit connaitre l’identificateur, d’où pour un identificateur on a :
  + Code identificateur (7 sur l’exemple)
  + Sa représentation (Chaine de caractères ou mieux, le numéro lexical[[2]](#footnote-2) : plus efficace)

1. Il s’agit d’un entier [↑](#footnote-ref-1)
2. Ou numéro dans le lexique [↑](#footnote-ref-2)