



# THÉORIES DES LANGAGES

**Mr,HEMIOUD**

**hemourad@yahoo,fr**

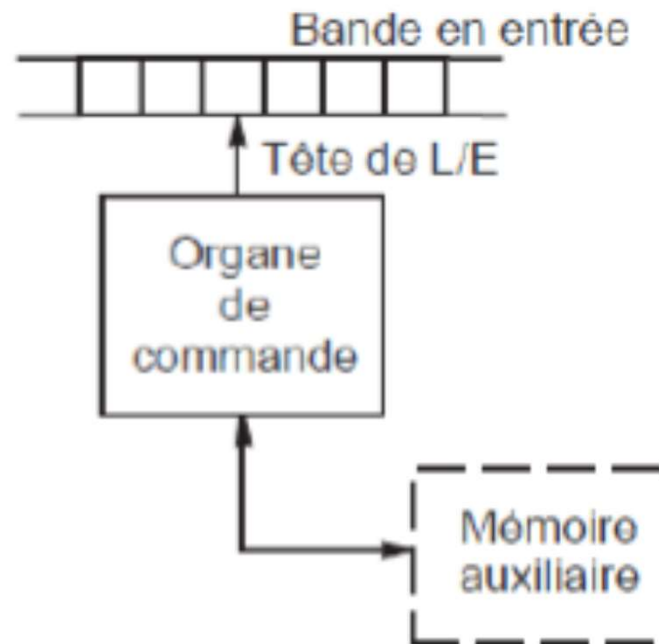
**Université de Jijel**

**Département d'informatique**

# LES AUTOMATES A PILES

# LES AUTOMATES À ÉTATS FINI

- Un **automate** est une machine abstraite qui permet de lire un mot et de répondre à la question : "**un mot  $w$  appartient-il à un langage  $L$  ?**" par **oui** ou **non**.
- Un automate est composé de :



## Limite des automates finis

- Certains langages ne peuvent pas être reconnus par les automates finis (ne peuvent être générés par une grammaire régulière)

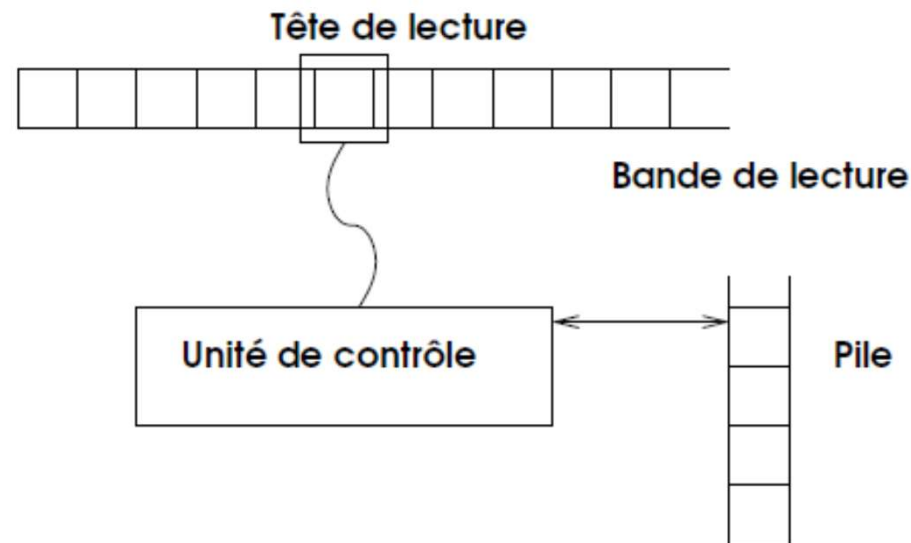
**Exemple** :  $L = \{a^n b^n \mid n \geq 0\}$

- Il faut *mémoriser* le nombre de *a* que l'on a lu pour vérifier que le mot possède autant de *b*.
- Pour mémoriser un nombre potentiellement infini de *a*, il faut un ensemble *infini* d'états !

# AUTOMATES À PILE

**Principe** : Les AuP fonctionnent sur le même principe que les AEF : depuis un état  $p$ , ils consomment un caractère du mot et effectue la transition correspondante qui les amène dans un nouvel état  $q$

À la différence des AEF, à chaque transition ils mettent à jour une pile et peuvent ainsi enregistrer des informations utiles pour la reconnaissance.



# Généralités

- Forme simple de *mémoire* : une **pile**.
  - Mode de stockage **Last In First Out**.
  - on accède à la pile **uniquement** par son sommet
- Deux *opérations* possibles :
  - **empiler** : ajouter un élément au sommet.
  - **dépiler** : enlever l'élément se trouvant au sommet.
- La pile permet de stocker de l'information **sans** forcément *multiplier* le *nombre d'états*.

# Configurations

## ○ Configuration initiale

- L'unité de contrôle est dans un **état initial**
- La tête est au début de la bande
- La mémoire contient un **élément initial**.( la pile est vide)

## ○ Configuration d'acceptation

- L'unité de contrôle est dans un **état d'acceptation**
- La tête de lecture est à **la fin** de la bande
- La mémoire se trouve dans un état d'acceptation ( la pile est vide)

**Un automate à pile** non-déterministe (APN) est un septuple  $(Q, A, P, \delta, q_0, Z, Q_F)$  avec :

- $Q$  : ensemble fini d'états
- $A$  : alphabet fini des symboles d'entrée
- $P$  : alphabet fini des symboles de pile (a priori  $P \cap A = \emptyset$ )
- $q_0$  : état initial
- $Z \in P$  : symbole initial de pile
- $Q_F \subseteq Q$  : ensemble des états terminaux
- $\delta$  est l'ensemble des règles de transition



- une règle  $\delta(\mathbf{p}, \mathbf{a}, \alpha) = (\mathbf{q}, \beta)$  de transition considère :
  - l'état courant  $\mathbf{p}$  de l'automate
  - le caractère lu  $\mathbf{a}$  sur le ruban (ou peut-être pas :  $\epsilon$ )
  - le symbole  $\alpha$  de **sommet** de pile (ou peut-être pas :  $Z$ )
- une **règle** indique :
  - le prochain état  $\mathbf{q}$  de l'automate
  - la suite de symboles  $\beta$  à **empiler** à la place du **sommet** de pile

# Configurations et mouvement

- **Configuration** :  $(q, w, \alpha) \in Q \times A^* \times P^*$  où :
  - $q$  représente l'état courant de l'unité de contrôle
  - $w$  est la partie du mot à reconnaître non encore lue. Le premier symbole de  $w$  (le plus à gauche) est celui qui se trouve sous la tête de lecture. Si  $w = \varepsilon$  alors tout le mot a été lu.
  - $\alpha$  représente le contenu de la pile. Le symbole le plus à gauche est le sommet de la pile. Si  $\alpha = Z$  alors la pile est *vide*.

- Configuration **initiale** :  $(q_0, w, Z)$  où  $w$  est le mot à reconnaître
- Configuration **d'acceptation** :  $(q, \varepsilon, Z)$  avec  $q \in Q_F$
- **Mouvement** :
  - $(p, aw, B) \vdash (q, w, AB)$  ( si  $\delta(p, a, B) = (q, AB)$  )
  - $(p, aw, AB) \vdash (q, w, B)$  ( si  $\delta(p, a, A) = (q, \varepsilon)$  )
  - $(p, aw, A) \vdash (q, w, A)$  ( si  $\delta(p, a, B) = (q, A)$  )

- **Représentation graphique**