

Université Bordeaux 1 — Master Informatique  
UE INF 316  
Traitement Automatique des Langues  
Examen du mardi 9 mai, 14h–17h  
Sujet de 4 pages  
documents autorisés / calculatrice bienvenue

**Exercice A Langage formel et langage naturel**

(A.i) En quoi le langage engendré par une grammaire formelle de type hors-contexte (encore dite non contextuelle ou algébrique) peut-il être considéré comme une formalisation de la performance linguistique d'un locuteur d'une langue donnée ?

**Exercice B Grammaires, automates et expressions régulières**

On considère la grammaire suivante,  $G$  dont les terminaux sont *anne*, *bernard*, *regarde*, *qui* et dont les non-terminaux sont  $S$ ,  $SN$ ,  $SV$ ,  $NP$ , et on note  $L(G)$  le langage qu'elle engendre.

$$\begin{array}{ll} S & \rightarrow SN\ SV \\ SN & \rightarrow NP \quad | \quad NP\ qui\ SV \\ SV & \rightarrow VT\ SN \\ NP & \rightarrow anne \quad | \quad bernard \\ VT & \rightarrow regarde \end{array}$$

(B.i) Quelle est le type de  $G$  (régulière, hors-contexte, contextuelle, générale) ?

(B.ii) Donner un automate qui engendre  $L(G)$ .

(B.iii) Quelle est le type le plus spécifique d'une grammaire engendrant  $L(G)$  parmi ceux-ci : régulière, hors-contexte, contextuelle, générale ?

(B.iv) Donner un automate qui reconnaisse l'intersection de  $L(G)$  avec le langage décrit par l'expression régulière suivante, dans la quelle  $ab$  est une abréviation pour l'expression régulière ( $anne \mid bernard$ )

$((qui\ regarde\ (ab\ qui\ regarde)^*) \mid \epsilon)\ anne\ regarde\ (ab\ qui\ regarde)^*\ bernard\ (qui\ regarde\ ab)^*$

### Exercice C Transducteurs

En italien, les noms singulier se finissent (sauf exception dont on ne tiendra pas compte) par *o*, *e*, *i*, *a* et leur pluriel se forme ainsi: les terminaisons *o*, *e*, *i* deviennent *i* et la terminaison *a* devient *e* mais pour les finales en *co*, *ca*, *go*, *ga* un *h* s'intercale entre la dernière consonne et la voyelle finale et leurs pluriels sont respectivement *chi*, *che*, *ghi*, *ghe* (pour préserver la prononciation de la consonne). Exemples:

- (1) libro  $\rightarrow$  libri
- (2) casa  $\rightarrow$  case
- (3) ponte  $\rightarrow$  ponti
- (4) banco  $\rightarrow$  banchi
- (5) barca  $\rightarrow$  barcha
- (6) albergho  $\rightarrow$  alberghi
- (7) alga  $\rightarrow$  alghe

(C.i) Ecrire un transducteur qui, étant donné, un nom italien au singulier (par exemple *libro*+*S*) rende ce nom au pluriel (*libri*+*P*) et qui ne modifie pas les mots que ne se finissent pas par +*S*.

### Exercice D Modèles de Markov Cachés

Le tableau ci-dessous donne les catégories syntaxiques des mots *le*, *la*, *paysan*, *ferme*.

	déterminant	adjectif	nom	pronom	verbe
le	x			x	
la	x			x	
paysan			x		
ferme		x	x		x

On suppose que

- une phrase ne peut commencer que par un article,
- une phrase ne peut pas finir par un pronom, ni par un article,
- les seules successions possibles de catégories sont les suivantes:
  - déterminant  $\rightarrow$  adjectif
  - déterminant  $\rightarrow$  nom
  - adjectif  $\rightarrow$  nom
  - adjectif  $\rightarrow$  verbe
  - adjectif  $\rightarrow$  pronom
  - nom  $\rightarrow$  adjective
  - nom  $\rightarrow$  verbe
  - nom  $\rightarrow$  pronom
  - pronom  $\rightarrow$  verbe
  - verbe  $\rightarrow$  déterminant

(D.i) Construire un Modèle de Markov Caché (HMM) avec état initial et état final en supposant qu'il y a équi-probabilité entre les transitions possibles et que la probabilité d'émettre chaque mot, pour une catégorie donnée, est la même.

(D.ii) Pour la phrase suivante, trouver tous les étiquetages syntaxiques différents et grammaticalement corrects (il y en a au moins deux) :

(8) Le paysan ferme la ferme.

(D.iii) Calculer la probabilité de l'étiquetage où "la" est un pronom (algorithme forward).

(D.iv) Quel est l'étiquetage donné par le Modèle de Markov Caché? (chemin de Viterbi)

(D.v) Réestimer les paramètres par la méthode de Viterbi à partir des exemples suivants:

(9) Le paysan ferme la ferme. (déjà traité)

(10) Le paysan la ferme.

(11) Le paysan ferme ferme la ferme.

### Exercice E Analyseur – Reconnaisseur

(E.i) Quelle est la différence entre un analyseur syntaxique et un reconnaisseur syntaxique ?

(E.ii) Décrire une opération de traitement de la langue qui nécessiterait un reconnaisseur sans nécessiter un analyseur.

### Exercice F Analyseur CKY

Décrire l'algorithme d'analyse syntaxique CKY. Nous demandons une explication rédigée qui, sans être technique, permettra à un novice de savoir comment fonctionne l'algorithme. Pour s'aider, on pourra répondre méthodiquement à ces questions :

- Que prend l'algorithme comme donnée initiale ?
- Que fournit-il en sortie d'une analyse ?
- Quelles sont les structures de données qu'il exploite ?
- Comment débute l'algorithme ?
- Etant donné un état d'avancement de l'algorithme, comment passe-t-il à l'étape suivante ?

### Exercice G Analyseur Earley

Soit la grammaire  $G = (\Sigma, N, R, S)$ , une grammaire indépendante du contexte avec

- $\Sigma = \{l', \text{obstination}, \text{de}, \text{cet}, \text{homme}, \text{brave}, \text{la}, \text{tourmente}\}$
- $N = \{P, GN, N, GV, GP, DET, NC, PREP, PRO, V\}$

- $R = \{$ 
  - $P \rightarrow GN\ GV,$
  - $GN \rightarrow DET\ N,$
  - $N \rightarrow NC,$
  - $N \rightarrow N\ GP,$
  - $N \rightarrow N\ ADJ,$
  - $GV \rightarrow V\ GN,$
  - $GV \rightarrow PRO\ V,$
  - $GP \rightarrow PREP\ GN$
  - $DET \rightarrow l'$
  - $NC \rightarrow obstination$
  - $PREP \rightarrow de$
  - $DET \rightarrow cet$
  - $NC \rightarrow homme$
  - $ADJ \rightarrow brave$
  - $V \rightarrow brave$
  - $DET \rightarrow la$
  - $PRO \rightarrow la$
  - $NC \rightarrow tourmente$
  - $V \rightarrow tourmente$
  - $\}$
- $S = P$

**(G.i)** Décrire en détail les étapes de l'algorithme de reconnaissance Earley qui permet de conclure à l'appartenance de la phrase "*L'obstination de cet homme brave la tourmente*" au langage  $L_G$  engendré par  $G$ .