

Examen de théorie des langages et compilation

Durée : 90 mn

Documents, instruments de calcul et téléphones non autorisés

Il sera tenu compte de la rigueur et de la clarté de la rédaction. Une pré

Exercice 1: Automates d'états finis (temps suggéré : 15mn) 3 points

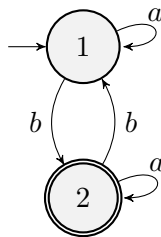
- (a) (1 point) Construire un AFD qui reconnaît le langage :
 $L_1 = \{w|w \text{ commence avec un 1 et se termine avec un 00}\}.$
- (b) (1 point) Construire un AFD qui reconnaît le langage :
 $L_2 = \{w|w \text{ contient au moins trois 0}\}.$
- (c) (1 point) Construire un AFD qui reconnaît le langage :
 $L_4 = \{w|w \text{ a une taille d'au plus 5}\}.$

Exercice 2: Langage des mots binaires multiples de 7 (temps suggéré : 20mn)..... 3 points

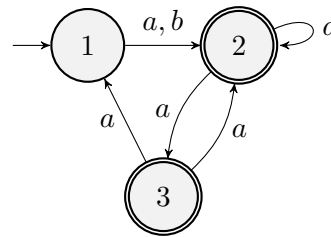
- (a) (3 points) Donner un automate fini déterministe pour reconnaître les mots en binaires qui sont multiples de 7. Exemples : $0, 111 = 7_d, 1110 = 14_d, 10101 = 21_d, \dots$

Exercice 3: AFD associé à une expression régulière (temps suggéré : 20mn) 4 points

- (a) (4 points) Convertir les deux automates suivants en expressions régulières équivalentes.



(a)



(b)

Exercice 4: Langages hors contexte (temps suggéré : 35mn) 10 points

Soit la grammaire suivante pour définir des expressions arithmétiques :

$$\begin{array}{lcl}
 S' & \rightarrow & S\$ \\
 S & \rightarrow & E \\
 E & \rightarrow & E + T \quad | \quad T \\
 T & \rightarrow & T * F \quad | \quad F \\
 F & \rightarrow & (E) \quad | \quad a
 \end{array}$$

- (a) (2 points) Calculer *premier* et *suivant* pour chaque non terminal.
- (b) (2 points) Calculer *directeur* de chaque règle.
- (c) (1 point) Cette grammaire est-elle LL(1) ? Justifier.
- (d) (3 points) Proposer des modifications de cette grammaire pour qu'elle devienne LL(1).
- (e) Dérouler l'algorithme de parsing LL(1) et donner les dérivations correspondantes sur les chaînes suivantes :
 - (a) (1 point) $(a * a) + a .$
 - (b) (1 point) $(a) + .$

Bon courage !