

ING1 - Théorie des langages - TP

13 Mai 2008

Dans ce TP vous allez découvrir l'utilitaire JFLAP (que vous trouverez sur AREL dans la section Outils). JFLAP permet de créer des automates d'états finis et de les tester.

1. Construire un automate déterministe qui reconnaît l'entrée du mot de passe d'un digicode. Il n'y a que des chiffres possibles en entrée. Le code est 34127.
2. Construire un automate déterministe reconnaissant le langage formé des mots dans lesquels toute occurrence de b est immédiatement suivie d'au moins deux occurrences de a .
3. Construire un automate déterministe reconnaissant le langage formé des mots qui ne contiennent pas deux a successifs.
4. Construire un automate déterministe reconnaissant le langage formé des mots dont les blocs de a sont alternativement de longueur paire et impaire.
5. Construire un automate déterministe reconnaissant le langage formé des mots contenant le facteur aba .
6. Construire un automate déterministe reconnaissant le langage formé des mots terminant soit par le suffixe aa soit par le suffixe abb .
7. Soit le langage L composé du seul mot *maman*:
 - Construire un automate déterministe reconnaissant l'ensemble des suffixes de L
 - Construire un automate déterministe reconnaissant l'ensemble des préfixes de L
 - Construire un automate déterministe reconnaissant l'ensemble des sous-mots de L
8. Construire un automate fini déterministe reconnaissant le langage $(a + b)^*(aa + bb)(a + b)^*$
9. Construire un automate fini déterministe reconnaissant le langage $((a^*bc^*)^*acbb^*)^*$

10. Construire un automate fini déterministe reconnaissant le langage $(a^*b^*)^*$
11. Construire un automate fini déterministe reconnaissant le langage $b(ab)^* + (ba)^*b$
12. Construire un automate fini déterministe reconnaissant le langage $(a + bb)^*(b + aa)^*$
13. Construire un automate fini (non déterministe) à $n + 1$ états reconnaissant le langage $(a + b)^*a(a + b)^{n-1}$