



## Travail 01

- Le travail est à **rendre avant le lundi 25 février à 16h00**.
  - Le travail est à effectuer en équipe de deux étudiants
  - La note est **individuelle** (déterminée lors de la présentation de la question 05)
- 

### Question 01 (10pts)

1. Donner 3 chaînes qui appartiennent à  $\{(012|1200|210)^*\}$  et à  $\{((200)^* | (01)^* | (21)^*)^*\}$  **(2pts)**
2. Modifier l'expression régulière  $r_0 = 0^*1^*$  pour qu'elle génère toutes les chaînes, appartenant à  $r_0$ , de longueur impair **(2pts)**
3. Donner l'expression régulière,  $\Sigma = \{0,1\}$ , qui permet de générer les chaînes qui ne contiennent pas de 00 ou 11. **(2pts)**
4. Donner l'expression régulière qui permet de générer : **(4pts)**
  - a. Les chaînes contenant « RIMOUSKI » ex : A**R**IN**M**OIP**U**SK**O**PIT
  - b. Les URL sachant que le domaine est 2 ou 3 caractères (fr, com...):  
ex : <http://www.a16a.fr>  
<https://aa.com>  
<http://www.a16a.cs.es>

### Question 02 (4pts)

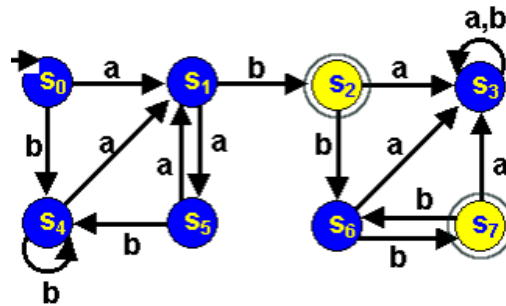
Donner le DFA qui accepte les chaînes qui ne contiennent pas aabb ( $\Sigma = \{a,b\}$ )

### Question 03 (8pts)

1. Donner le NFA qui correspond à l'expression régulière suivante : **(4 pts)**  
 $(a|b)^*(abb|a^+b)$
2. Donner le DFA de l'automate précédent **(4 pts)**

### Question 04 (8pts)

1. Donner l'expression régulière correspondante au DFA suivant : (4 pts)



2. Minimiser le. (4 pts)

### Question 05 (20pts)

Créer votre propre **Lexer** qui permet, en lisant un fichier texte contenant un programme écrit en c#, de :

- Trouver les erreurs lexicales, s'il y en a
  - D'attribuer chaque lexème au "token" que vous avez défini au préalable (pdf BNF diapo 9)
- Ne gérer que :
- Déclaration : int, float, bool, char, string
  - Identificateur : alphanumérique, commençant par une lettre et finissant par une lettre
  - Condition : if
  - Opérateur : ==, !=, <, >
  - Boucle : for
- D'afficher le contenu de la table de symboles. (au stade de l'analyse lexicale)