# MAT115 - Devoir #5

À remettre le **lundi 15 avril avant 23h59**.
Tout retard entraînera jusqu'à 33% de pénalité par jour.

**Pondération.** Ce travail compte pour 7.5% des points de la session.
**Modalités.** Considérez les points suivants.

- Le travail peut être fait seul ou en équipe de deux.
- Le devoir doit être remis en format pdf via turnin. La provenance de votre pdf n'a pas d'importance — ce pdf peut provenir d'un export d'un fichier Word ou d'un fichier texte, d'un scan de vos écrits/dessins, etc.

  Il n'y a pas de norme de présentation particulière. Écrivez lisiblement, et assurez-vous que vos noms et CIP apparaissent clairement sur le devoir remis.
- Vous pouvez utiliser les techniques de preuves que vous désirez, les indices ne sont que des suggestions.
- Sauf indication contraire, vous pouvez utiliser les résultats démontrés en classe et dans les solutionnaires d'exercices sans avoir à les démontrer (par exemple, par exemple, vous pouvez utiliser les tables des lois de la logique). Par contre, si vous utilisez de nouveaux résultats, vous devez les démontrer.

**Question 1 : création d'automates** ($8+8+8+8+8=40$ **points)**

Pour chacun des énoncés ci-bas, donnez un automate dont le langage est l'ensemble de mots spécifié. Vous pouvez remettre un dessin de votre automate ou bien la définition formelle de son quintuplet. Sauf indication contraire, votre automate peut être un AFD ou un AFND.

a. Les codes postaux canadiens, qui sont formés de 6 ou 7 symboles. Quand on lit un code postal de gauche à droite, on doit lire: une lettre, un chiffre, une lettre, un espace *optionnel*, un chiffre, une lettre, puis un chiffre. Par exemple, J1K 2R1 ou J1K2R1 sont des formats corrects.

   On peut représenter l'espace par ␣. L'alphabet est $[A-Z] \cup [0-9] \cup \{␣\}$.

b. Les mots sur alphabet $\{0,1\}$ qui ont un nombre pair de 0 ou un nombre impair de 1.

c. Les mots sur alphabet $\{0,1\}$ qui n'ont *pas* un nombre pair de 0, et qui n'ont *pas* un nombre pair de 1.

d. Les mots sur alphabet $\{0,1\}$ qui contiennent au moins une occurrence de la sous-chaîne 00100. Votre automate doit être déterministe.

e. Une ligne de code C++ qui déclare une variable de type `int`, possiblement initialisée à un nombre entier. Le nom de variable est sur alphabet [a-z]. On peut ajouter un nombre arbitraire d'espaces entre les éléments de la ligne. Voici cinq exemples de lignes valides que vous devez accepter:

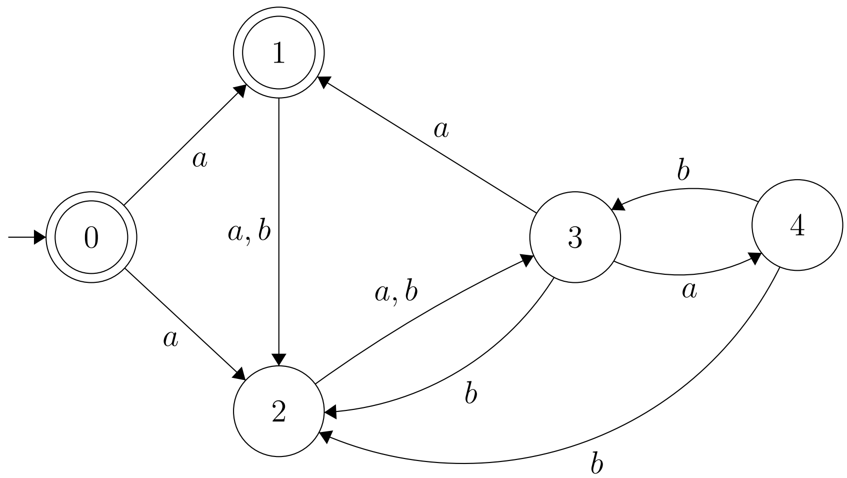
```
int x;
int x = 3;
int allo=3;
    int    salut  =    -3    ;
    int    bonjour    ;
```

   Notez l'ajout d'espaces dans les deux derniers exemples.

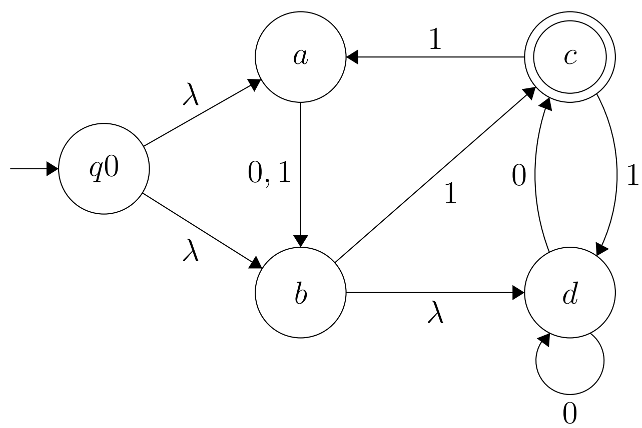
   Pour décrire en détail le format, une telle ligne est constituée, dans l'ordre, de: une série de 0 espaces ou plus (pour l'indentation), suivi du mot `int`, suivi de 1 espace ou plus, suivi du nom de la variable (qui ne contient que des symboles de [a-z]), suivi de 0 espaces ou plus. Ensuite, on peut soit ajouter le symbole ";" pour terminer la déclaration, ou ajouter "=" suivi de 0 espaces ou plus, puis d'un mot représentant un entier positif ou négatif (sans espaces), puis de 0 espaces ou plus, puis de ";". À vous de déterminer l'alphabet.

**Question 2 : manipulation d'automates** ($10 + 10 + 10 = 30$ **points)**

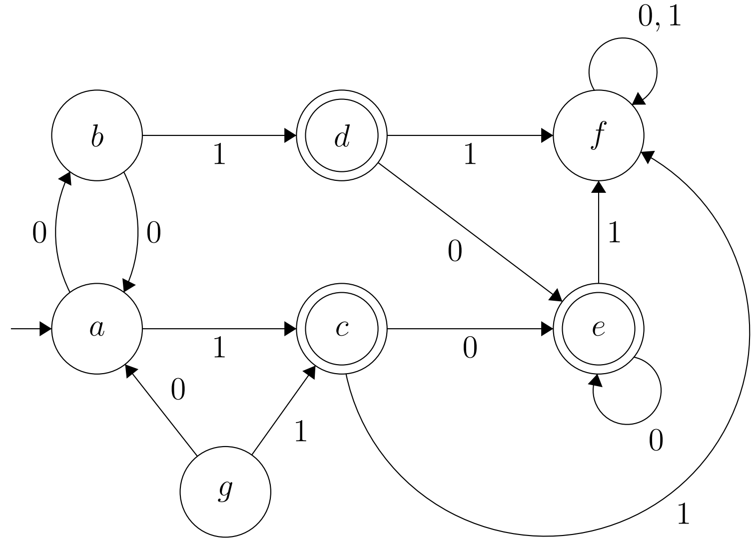
a. Transformez l'AFND suivant en un AFD équivalent.




b. Transformez l'AFND suivant en un AFND équivalent qui n'a pas de transition $\lambda$.



c. Minimisez l'AFD suivant. Dites dans quel ordre vous avez déterminé vos partitions en états indistinguables (sans détailler comment vous avez obtenu vos partitions).



**Question 3 : maths et automates (5 + 5 + 10 + 10 points)**

a. Soit $A = \langle Q, \Sigma, \delta, q_0, F\rangle$ un AFND. Soit un AFD $A' = \langle Q', \Sigma', \delta', q_0', F'\rangle$ tel que $L(A) = L(A')$ obtenu avec les approches vues en classe. Expliquez pourquoi il est vrai que $|Q'| \leq 2^{|Q|}$.

b. Soit $A = \langle Q, \Sigma, \delta, q_0, F\rangle$ un AFND. Montrez que $|\delta| \leq |Q|^2 \cdot |\Sigma|$.

c. Un ensemble de mots $M \subseteq \Sigma^*$ est appelé *régulier* s'il existe un AFD ou un AFND $A$ tel que $L(A) = M$. Montrez que si $M_1, M_2 \subseteq \Sigma^*$ sont deux ensembles de mots réguliers, alors $M_1 \cup M_2$ est régulier.

   *Suggestion.* Je recommande de décrire un automate dont le langage est $M_1 \cup M_2$. Vous devez argumenter que le langage de votre automate est bel et bien le bon.

d. Soit $M_1, M_2 \subseteq \Sigma^*$ deux ensembles de mots réguliers. Donnez un AFD ou un AFND dont le langage est $M_1 \setminus M_2$.

   Il n'est pas exigé d'argumenter que votre automate est correct - il vous suffit de décrire votre automate. Je recommande de vous inspirer des exercices, où on montre que $M_1 \cap M_2$ est régulier.