

NFP 108: feuille d'exercices numéro 2

F. Barthélemy et O. Pons

23 octobre 2007

Exercice 1

Donnez sous forme graphique les automates finis correspondant aux expressions régulières suivantes :

- a^*
- a^*bc^*
- $ab(a|b)^*$
- $ab((a|b)^*c)^*$

Exercice 2

1. donnez une expression régulière très simple permettant de décrire les nombres entiers positifs ou nul, en représentation décimale (base 10).
2. donnez une expression régulière représentant ces mêmes nombres en interdisant que le premier chiffre soit 0.
3. modifiez l'expression précédente pour représenter les nombres entiers positifs, négatifs ou nul.
4. donnez une expression régulière représentant les nombres décimaux.
5. donnez une expression des nombres représentant les montants en euros et centimes.

Exercice 3

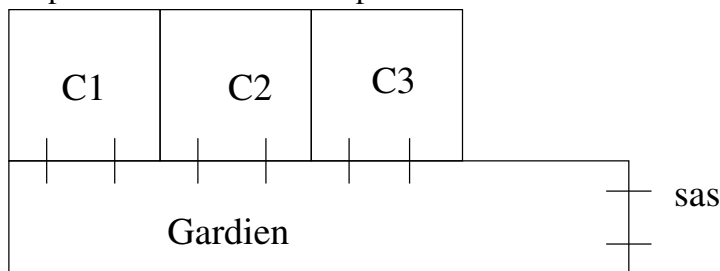
Dans cet exercice, on va chercher à décrire les lexèmes (en anglais, token) d'un langage de programmation au moyen d'expressions régulières. Le langage en question sera un petit sous-ensemble de java, avec les constructions suivantes :

- les types `int`, `String`, `void`, `boolean`.
- les constantes des types en questions (nombres, chaînes entre guillemets, `true` et `false`).

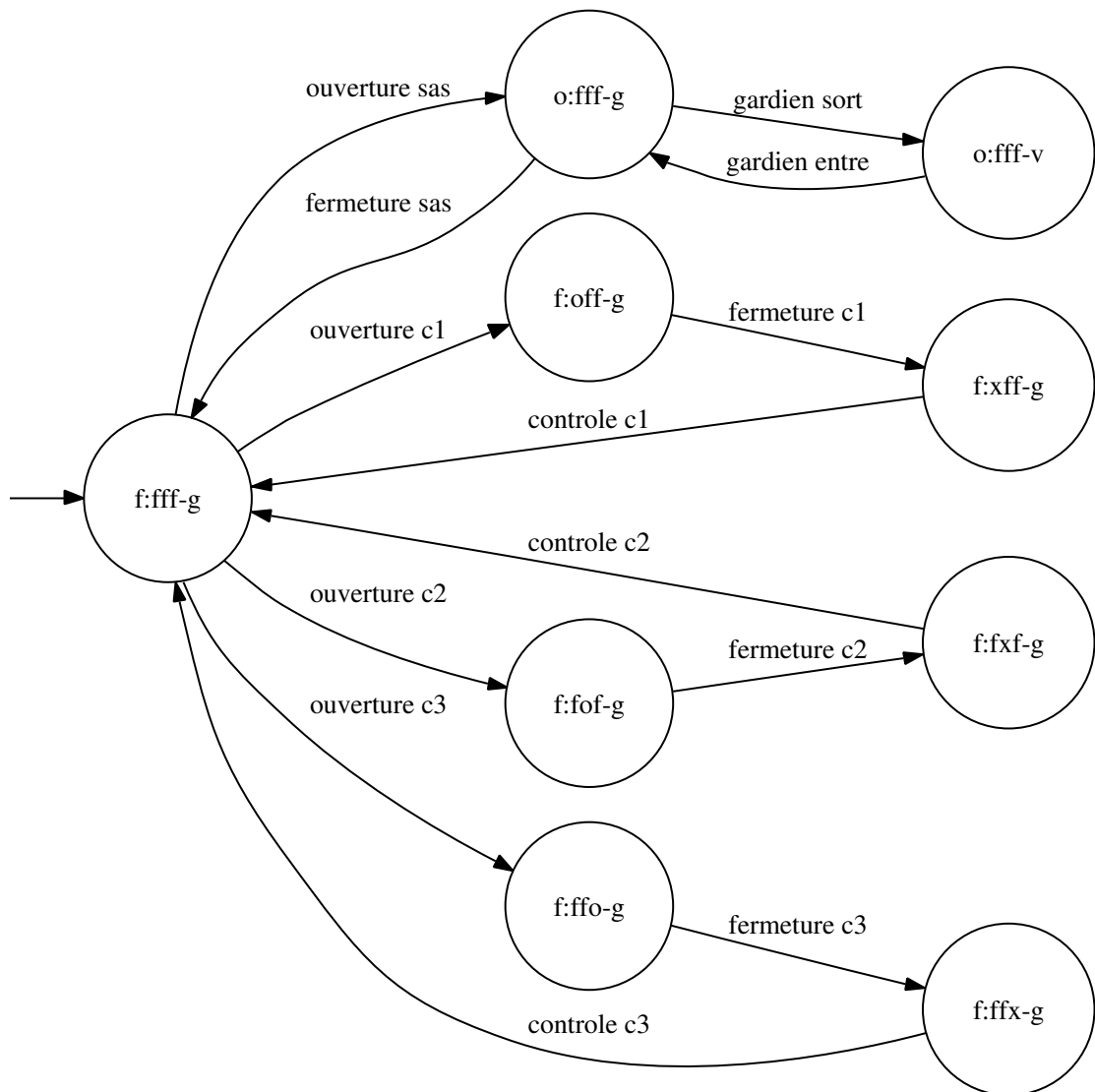
- les instructions seront : des envois de messages, des affectations, des if, des déclarations de variables simples. Les expressions utiliseront des opérateurs simples, arithmétiques et booléens.
 - les identificateurs en java commencent par une lettre suivie de lettres ou chiffres.
1. donnez une expression régulière décrivant les types
 2. donnez une expression régulière décrivant les valeurs
 3. donnez une expression régulière décrivant les symboles spéciaux comme le point-virgule ou le signe égal pour l'affectation.
 4. donnez un langage régulier décrivant les identificateurs (vous pourrez utiliser les résultats précédants en leur donnant un nom).
 5. donnez un langage régulier décrivant les mots-clés du langage
 6. donnez un langage régulier donnant l'ensemble des lexèmes du langage.
 7. donnez une expression régulière décrivant une déclaration de variables.

Exercice 4

Le plan d'un couloir d'une prison de haute sécurité :



Le couloir dessert trois cellules c1, c2 et c3. Chacune a une porte. Il y a également une porte en bout de couloir. L'ouverture et la fermeture des portes est commandé à distance depuis un PC de sécurité qui dispose en plus de caméras vidéo pour les différents lieux. La fermeture d'une cellule se fait en deux temps : l'actionnement de la cellule, puis un contrôle visuel par la vidéo que la cellule contient bien le détenu ou qu'elle est bien vide si elle doit l'être. La logique du mécanisme est décrite par l'automate est donné ci-dessous. Les noms d'état sont composés avec d'abord l'état de la porte du couloir, deux points, puis les états des trois cellules et enfin la présence ou absence d'un gardien dans le couloir. Les états sont : f pour fermé et contrôlé, o pour ouvert, x pour fermé non contrôlé, g pour la présence et v pour l'absence du gardien.



1. Peut-on ouvrir une porte s'il n'y a pas de gardien dans le couloir ?
2. Peut-on ouvrir plusieurs cellules en même temps ?
3. Décrivez l'arrivée d'un nouveau détenu dans la cellule 2, en supposant qu'elle est vide initialement. Précisez les états successifs.
4. A quoi sert l'état $o : f f f - v$ (en trois lignes) ?
5. En cas d'incendie, il faut changer les procédures de sécurité pour évacuer au plus vite le gardien et les détenus. Comment peut-on modifier l'automate pour prendre cela en compte ?
6. Lors de la fermeture d'une cellule, le comportement du système n'est pas satisfaisant pour ce qui est du contrôle visuel. Expliquez pourquoi et proposez une solution.
7. Cet automate est-il déterministe ?
8. Quel est le langage reconnu par cet automate ?