Langages formels

3. Fonctions séquentielles

21 février 2007

Exercice 1 – *Opérations sur les entiers*

Dans cet exercice on considère deux représentations des entiers en base 2 (ou plus généralement en base β) : la représentation *classique* (avec le chiffre de poids fort en premier) et la représentation *inverse* (avec le chiffre de poids faible en premier).

1. Donner un transducteur séquentiel qui réalise la multiplication par 3 en base 2 (représentation inverse).
2. Donner un transducteur séquentiel qui réalise la division entière par 3 en base 2 (représentation classique).
3. On définit l'opérateur comparaison comme la fonction prenant en argument deux entiers x et y en représentation inverse et qui renvoie $\top$ si $x \geqslant y$ et $\perp$ sinon. Donner un transducteur séquentiel qui réalise la comparaison (on suppose que les codages de x et y ont même longueur).
4. On définit la soustraction comme la fonction prenant deux entiers x et y en représentation inverse et qui renvoie l'entier $x - y$ en base 2 inverse si $x \geqslant y$ et le caractère # en dernier sinon. Donner un transducteur séquentiel qui réalise la soustraction.
5. Donner un transducteur séquentiel qui reconnaît l'ensemble des représentations classiques d'entiers vérifiant $2x + 3y \equiv 1\ [6]$. De même pour $\{\, x \mid \exists y.\, 2x + 3y \equiv 1\ [6] \,\}$.

Exercice 2 – *Codes à délai de déchiffrage fini*

1. Soit $A = \{a, b\}$, soit $\beta_1 : \{x, y\}^* \to A^*$ le morphisme défini par $\beta_1(x) = a$ et $\beta_1(y) = aba$. La relation β_1^{-1} est-elle une fonction séquentielle ?
2. Même question avec $\beta_2 : \{x, y, z\}^* \to A^*$ défini par $\beta_2(x) = ab$, $\beta_2(y) = abb$ et $\beta_2(z) = baab$.
3. Généralisation : Soit X un sous-ensemble fini de A^* ; X est dit *préfixe* si aucun mot de X n'est préfixe d'un autre mot de X. Soit $B = \{x_1, \dots, x_n\}$ un ensemble en bijection avec X ; cette bijection induit un morphisme $\beta : B^* \to A^*$. On dit que X est un *code* si ce morphisme est injectif. Un code X est dit *à délai de déchiffrage* d si, quand un mot $f = x_1 x_2 \dots x_{d+1} \in X^{d+1}$ est *préfixe* (en tant que mot de A^*) d'un mot $g = y_1 y_2 \dots y_r \in X^*$, alors on a $x_1 = y_1$.
   - **a.** Vérifier qu'un code préfixe est un code à délai de déchiffrage 0.
   - **b.** Donner un exemple de code qui n'est pas à délai de déchiffrage fini.
   - **c.** Montrer que si X est préfixe, β^{-1} est une fonction séquentielle pure.
   - **d.** Montrer que si X est un code à délai de déchiffrage fini, β^{-1} est une fonction séquentielle.
   - **e.** Réciproquement, montrer que si β^{-1} est une fonction séquentielle, alors X est un code à délai de déchiffrage fini.

On a démontré la proposition suivante :

> Soit $\beta : B^* \to A^*$ un morphisme. L'ensemble $X = \beta(B)$ est un code à délai de déchiffrage fini si et seulement si β^{-1} est une fonction séquentielle.