TD 02 – Cardinalité et machines de Turing et codage

Rappel : soient A et B deux ensembles, une fonction $f : A \to B$ est
— injective ssi $\forall\, a, a' \in A : f(a) = f(a') \implies a = a'$ (ou la contraposée),
— surjective ssi $\forall\, b \in B : \exists\, a \in A : b = f(a)$,
— bijective ssi elle est à la fois injective et surjective.

Utile : Théorème de Cantor-Schröder-Bernstein : soient A et B deux ensembles, si il existe une fonction injective de A vers B (intuitivement $|A| \leq |B|$), et une fonction injective de B vers A (intuitivement $|B| \leq |A|$), alors il existe une bijection entre A et B (intuitivement $|A| = |B|$).

Exercice 1. *Ensembles infinis dénombrables*

1. Donner cinq éléments de l'ensemble $\mathbb{N} \times \{0, 1, a\}$.
2. Montrer que $\mathbb{N} = \{0, 1, 2, \ldots\}$ et $2\mathbb{N} = \{0, 2, 4, \ldots\}$ sont en bijection.
3. Montrer que $\mathbb{N} = \{0, 1, 2, \ldots\}$ et $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$ sont en bijection.
4. Montrer que $\mathbb{N}$ et $\mathbb{N} \times \mathbb{N}$ sont en bijection.
5. Peut-on en déduire que $|\mathbb{Q}| = |\mathbb{N}|$?
6. Montrer que $\mathbb{N}$ et $\mathbb{N} \times \mathbb{N} \times \mathbb{N}$ sont en bijection.
7. Montrer que Σ^* et $\mathbb{N}$ sont en bijection, pour Σ un alphabet fini.

Exercice 2. *Ensembles infinis indénombrables*

1. Montrer que $\mathcal{P}(\mathbb{N})$ (l'ensemble des parties de $\mathbb{N}$) et $[0, 1]$ sont en bijection.
2. Montrer que l'ensemble des langages sur un alphabet fini Σ, et $[0, 1]$, sont en bijection.
3. Montrer que $[0, 1]$ et $\mathbb{R}$ sont en bijection.

Exercice 3. *Codage*

On définit une fonction de codage $code_1$ des couples d'entiers de la manière suivante : étant donnés deux entiers x et y, on pose

$$code_1(x, y) = x_1 0 x_2 0 \ldots x_{n-1} 0 x_n \, 1 \, y_1 y_2 \ldots y_m,$$

où $x_1 x_2 \ldots x_n$ et $y_1 y_2 \ldots y_m$ sont les représentations binaires respectives de x et y.

1. Ce codage est-il injectif ? (Autrement dit, un tel code correspond-il bien à un unique couple d'entiers ?)
2. Donnez le codage du couple $(10, 5)$.
3. Quel est le couple associé au codage 100010100010111001110 ?
4. Quel est le nombre de bits utilisés pour coder un couple (x, y) ?

On définit un nouveau codage $code_2$ en posant maintenant

$$code_2(x, y) = t_1 0 t_2 0 \ldots t_{\ell-1} 0 t_\ell \, 1 \, x_1 x_2 \ldots x_n \, y_1 y_2 \ldots y_m,$$

où $x_1 x_2 \ldots x_n$ et $y_1 y_2 \ldots y_m$ sont les représentations binaires respectives de x et y, et où $t_1 t_2 \ldots t_\ell$ est la représentation binaire de l'entier n (qui est lui-même la taille de la représentation binaire de x).

5. Ce codage est-il injectif ?
6. Donnez le codage du couple $(10, 5)$.

7. Quel est le couple (x, y) associé au codage 100011100011001 ?
8. Quel est le nombre de bits utilisés pour coder un couple (x, y) ?

On souhaite généraliser ce codage aux tuples d'entiers $(x^1, \ldots, x^k)$ pour k quelconque.

9. Proposez un tel codage.
10. Donnez le codage du tuple $(6, 11, 10, 3)$.
11. Donnez le nombre de bits utilisés pour coder un tuple arbitraire $(x^1, \ldots, x^k)$.

Exercice 4. *MT : conventions*

Objectif : voir que l'on peut utiliser d'autres conventions.

1. Peut-on décider exactement les mêmes langages si la tête de lecture est initialement placée sur la case la plus à droite du mot d'entrée ? (Justifier)
2. Peut-on décider exactement les mêmes langages si l'on autorise des transitions pour lesquelles la tête de lecture/écriture ne bouge pas (ni L ni R) ? (Justifier)
3. Peut-on décider exactement les mêmes langages si l'on ajoute la restriction $\Sigma = \{0, 1\}$? Et si en plus $\Gamma = \{0, 1, B\}$? (Justifier)

Exercice 5. *Espace et temps*

Supposons qu'une machine de Turing s'arrête au bout de t étapes de calcul en visitant (possiblement plusieurs fois chacune) s cases du ruban. On a évidement $s \leq t$ car au plus une nouvelle case est visitée par étape de temps, mais peut-on borner supérieurement t par une fonction de s ?

Exercice 6. *MT et pseudo-code*

Soit le pseudo-code suivant.

```
h(x : tableau de bits de taille n)
  i <- n
  tant que ( x[i] == 1 et i >= 1 ) faire
    i <- i-1
  fin tant que
  si i == 0 alors
    accepter
  sinon
    rejeter
  fin si
```

1. Quel est le langage reconnu/accepté par l'algorithme h ?
2. Donner une machine de Turing qui décide le même langage que h, et qui s'arrête toujours.
3. Donner une machine de Turing qui calcule la fonction $f : \mathbb{N} \to \mathbb{N}$ définie par :

$$f(n) = \begin{cases} n/2 & \text{si } n \text{ est pair,} \\ n & \text{sinon.} \end{cases}$$

4. Donner une machine de Turing qui calcule la fonction $g : \mathbb{N} \to \mathbb{N}$ telle que $g(n) = 2^n$ pour tout $n \in \mathbb{N}$.
5. Sauriez-vous construire une machine de Turing pour convertir un entier codé en binaire en un entier de même valeur codé en unaire ?