Planche TD 1 - Calculabilité

Ivan Lejeune*

13 février 2024

1 Divers

Exercice 1 - Ensemble infini. On considère les problèmes suivants :

- Hôtel d'Hilbert : Un hôtel possède une infinité de chambres.
  Peut-on trouver une chambre libre pour un nouveau client ?
- Hôtel d'Hilbert suite : Un hôtel possède une infinité de chambres.
  Peut-on trouver une infinité de chambres libres pour une infinité de clients ?

1. Proposer une solution pour les deux problèmes.
2. Modéliser les deux problèmes et proposer pour chaque problème une fonction mathématique de $\mathbb{N}$ dans $\mathbb{N}$.

Exercice 2 - Paradoxe. Montrer que les problèmes suivants engendrent un paradoxe :

1. Le conseil municipal d'un village vote un arrêté municipal qui enjoint à son barbier (masculin) de raser tous les habitants masculins du village qui ne se rasent pas eux-mêmes et seulement ceux-ci.
2. Un crocodile s'empare d'un bébé et dit à la mère : "Si tu devines ce que je vais faire, je te rends le bébé, sinon je le dévore".
   En supposant que le crocodile tienne parole, que doit dire la mère pour que le crocodile rende l'enfant à sa mère ?
   Une réponse usuelle de la mère est "Tu vas le dévorer !".

2 Variations sur le codage

Exercice 3 - Codage de couples d'entiers. Soit la fonction suivante :

$$\begin{aligned} Rang : \quad \mathbb{N} \times \mathbb{N} &\to \quad \mathbb{N} \\ (x, y) &\mapsto \quad \frac{(x+y)(x+y+1)}{2} + x \end{aligned}$$

1. Calculer $Rang(4, 5)$. Donner le couple pour lequel la valeur du codage est 8.
2. Donner une version récursive de la fonction $Rang$.
3. Donner la fonction inverse.

Exercice. Soit c la fonction de codage pour les couples d'entiers vue en cours.

1. Soit h la fonction de codage pour les triplets définie par

$$h(x, y, z) = c(c(x, y), z)$$

*Cours inspiré de M. Giroudeau

Quel est le doublet codé par 67 ? Quel est le triplet codé par 67 ?

2. Le couple (z, t) succède au couple (x, y) si

$$c(z, t) = c(x, y) + 1$$

Écrire la fonction successeur qui prend en paramètre un couple et retourne le couple successeur.

Exercice. Proposer un codage pour les nombres rationnels. Un nombre rationnel r est caractérisé par une paire de naturels (a, b) telle que $r = \frac{a}{b}$, $b \neq 0$ et telle que a et b n'ont pas de facteurs communs.

Exercice. Pour coder les listes d'entiers, peut-on :

1. faire la somme des entiers de la liste, et à somme égale prendre l'ordre lexicographique ?
2. faire comme pour les mots : prendre les listes les plus courtes d'abord et à égalité prendre l'ordre lexicographique ?

Exercice. On ordonne les listes de la façon suivante :

$$\sigma(l) = \text{la somme des entiers de la liste} + \text{la longueur de la liste}$$

Puis à valeur de σ égale on ordonne dans l'ordre lexicographique.
On note U_k l'ensemble des listes l telles que

$$\sigma(l) = k \quad \text{et} \quad u_k = |U_k|$$

1. Donner les ensembles U_i pour $i \in [\![1, 4]\!]$.
2. Montrer que $u_k = 2^{k-1}$ pour tout $k \geq 1$.
3. Pour $k \geq 1$, quelle est la première liste de U_k ? Et la dernière ?
4. Donner la fonction de codage en version itérative et récursive (respectivement décodage).

Exercice. Soit la fonction f de $\mathbb{N}^* \to \mathbb{N}$ définie comme suit :

$$f(n) = \begin{cases} k & \text{si } n = 2^k \\ f(\frac{n}{2}) & \text{si } n \text{ pair et n'est pas une puissance de 2} \\ f(3n+1) & \text{sinon} \end{cases}$$

On appelle

$$A_i = \{x \mid f(x) = i\}$$

1. Donner quelques éléments de A_i pour $i \in [\![1, 6]\!]$.
2. Donner un algorithme qui prend i en paramètre et qui affiche tous les éléments de A_i.
3. Donner un algorithme qui affiche $A_1 \cup A_2$.
4. Donner un algorithme qui affiche $A_4 \cup A_6$

Exercice. Le but est de donner un numéro entier naturel à chaque objet manipulé (mot, instruction machine de Turing,etc...) et de remplacer des manipulations de ces objets par des opérations de nature arithmétique sur leurs numéros. Nous commençons par les séquences d'entiers. Le numéro de Gödel d'un n-uplet d'entiers naturel $(x_1, \ldots, x_n)$, noté par $\langle x_1, \ldots, x_n \rangle$, est défini comme suit :

$$\begin{aligned} A = \langle x_1, \ldots, x_n \rangle &= 2^n \cdot 3^{x_1} \cdot 5^{x_2} \cdots p_n^{x_n} \\ &= 2^n \prod_{i=1}^{n} p_i^{x_i} \end{aligned}$$

1. Donner $\langle 10, 11 \rangle$.
2. Le numéro de Gödel d'un n-uplet est une opération injective ? Surjective ?
3. Étendre le nombre de Gödel au mot défini par

$$w = a_{i_1} a_{i_2} \dots a_{i_l}$$

4. Étendre le nombre de Gödel aux instructions

$$I \ : \ q_i a_j \to q_k a_l \begin{Bmatrix} - \\ G \\ D \end{Bmatrix}$$

pour les machines de Turing.

> **Rappel.** Les instructions ont la forme suivante :
>
> (état, symbole lu) $\to$ (état, symbole écrit, déplacement).

5. Étendre le nombre de Gödel au numéro d'une machine de Turing avec $I_1, \dots, I_n$ les instructions de la machine M.
6. Étendre le nombre de Gödel au numéro d'une configuration

$$c = (w_G, q_i, w_D)$$

Exercice. Soit f une application de $\mathbb{N}$ dans $\mathbb{N}$ telle que :

$$\forall (m, n) \in \mathbb{N}, f(m^2 + n^2) = f(m)^2 + f(n)^2$$

On veut montrer que f est :

- l'application nulle, donnée par
$$\forall n \in \mathbb{N}, f(n) = 0$$
- l'application identité, donnée par
$$\forall n \in \mathbb{N}, f(n) = n$$

On supposera que a est l'entier naturel $f(1)$.

1. Montrer que $f(0) = 0$. En déduire que pour tout $n \in \mathbb{N}$, on a

$$f(n^2) = f(n)^2$$

2. Montrer alors que $a^2 = a$, donc que a est égal à 0 ou à 1. Pour répondre, il suffit de prouver que l'égalité
$$f(n) = an$$
est vrai pour tout $n \in \mathbb{N}$.
3. Vérifier successivement les égalités suivantes :

$$f(2) = 2a$$
$$f(4) = 4a$$
$$f(5) = 5a$$

4. Utiliser les valeurs de $f(4)$ et de $f(5)$ pour montrer que $f(3) = 3a$
5. Utiliser les valeurs de $f(1)$ et de $f(5)$ pour montrer que $f(7) = 7a$.

6. Montrer les égalités suivantes :
$$f(8) = 8a$$
$$f(9) = 9a$$
$$f(10) = 10a$$
$$f(6) = 6a$$
7. Observer que pour tout $m \in \mathbb{N}$, on a
$$\begin{cases} (2m)^2 + (m-5)^2 = (2m-4)^2 + (m+3)^2 \\ (2m+1)^2 + (m-2)^2 = (2m-1)^2 + (m+2)^2 \end{cases}$$
8. Montrer que pour tout $n \in \mathbb{N}$, on a $f(n) = an$.
9. Conclure.

3 Diagonalisation

Exercice. Montrer que $[0, 1[$ n'est pas dénombrable, puis conclure.

Exercice. On considère l'ensemble U des suites $(u_n)_{n\in\mathbb{N}}$ à valeurs dans $\{0, 1\}$. Montrer que U n'est pas dénombrable.

4 Dénombrabilité

Exercice. Un ensemble est **fini** si on ne peut pas le mettre en bijection avec une partie stricte de lui-même. Il est infini sinon.
Montrer que l'ensemble des entiers est infini.

Exercice. Montrer que l'ensemble F des parties finies de $\mathbb{N}$ est dénombrable.

5 Fonctions (non)-calculables

Exercice. Soit $f : \mathbb{N} \to \{0, 1\}$ une fonction totale non calculable.

1. Rappeler la définition d'une fonction totale et d'une fonction non calculable.
2. Construire une fonction $g_1 : \mathbb{N} \to \mathbb{N}$ totale, croissante et non calculable à partir de f.
3. Construire une fonction $g_2 : \mathbb{N} \to \mathbb{N}$ totale, strictement croissante et non calculable à partir de f.
4. Est-il vrai que toute fonction $f : \mathbb{N} \to \mathbb{N}$ qui est non bornée est également non calculable ?

Exercice 16 - Calculabilité.
1. Une fonction $\mathbb{N} \to \mathbb{N}$ totale, non calculable et croissante, peut-elle être bornée ?
2. Montrer que toute fonction totale, croissante et bornée est calculable. Que se passe-t-il si une fonction est totale décroissante ?

Exercice 17 - Calculabilité.
1. Montrer que l'inverse d'une fonction f calculable et bijective est calculable.
2. Si f n'est pas surjective, quel est le problème rencontré avec la procédure précédente ? Même chose pour une fonction surjective mais partielle.
3. Si f n'est pas injective, que calcule f^{-1} ?

Exercice. Montrer qu'une fonction f totale $\mathbb{N} \to \mathbb{N}$ est calculable si et seulement si son graphe

$$G = \{(x, f(x)) \mid x \in \mathbb{N}\}$$

est décidable.

Exercice. Soient E un ensemble et ϕ une fonction telle que $\phi(n)$ est égale au nombre d'éléments de E strictement inférieur à n.
Montrer que ϕ est calculable si et seulement si E est décidable.

6 Problèmes indécidables

Exercice. Nous considérons la fonction suivante donnée par l'algorithme 1 :

Algorithme 1 : La fonction de Collatz

```
Data : n ∈ ℕ
Result : k ∈ ℕ
while n ≠ 1 do
    if n = 0 (mod 2) then
        n ← n/2
    end
    else
        n ← 3 × n + 1
    end
end
```

Actuellement nous ne savons pas si cette fonction termine pour tout n.
Est-ce que vous êtes d'accord avec la preuve suivante ?

"Si le problème de l'arrêt était décidable, il suffirait de l'appliquer à ce programme pour savoir si son exécution s'arrête. Or, on ne sait pas si son exécution s'arrête. D'où la contradiction."

Exercice. Né en Pologne en 1897, Emil Post émigre pour les États-Unis en 1904, où il mène de brillantes études à l'université de Columbia. Malgré une maladie très handicapante, il conduit des recherches en logique au City College de New York qui l'amènent à définir en 1946 le problème qui porte aujourd'hui son nom, dont il démontre l'indécidabilité. Il meurt en 1954.
Le problème de *Post* est un problème de décision.