Feuille d'exercices n°1

Fonctions récursives primitives

Les exercices de cette feuille sont essentiellement du cours. On ne doit utiliser pour un exercice donné que les résultats des exercices qui précèdent.

Définitions. Posons $\mathcal{F} = \bigcup_{p\in\mathbb{N}^*} \mathbb{N}^{\mathbb{N}^p}$. L'ensemble des *fonctions récursives primitives* est le plus petit sous-ensemble de $\mathcal{F}$

i. qui contient les fonction *nulle* $\theta : \mathbb{N} \to \mathbb{N}$, la fonction *successeur* $s : \mathbb{N} \to \mathbb{N}$, les *projections* $p_k^i : \mathbb{N}^k \longrightarrow \mathbb{N}$ $(1 \le i \le k)$ définies par $p_k^i(x_1, \dots, x_k) = x_i$.

ii. qui est clos par le *schéma de composition* :
si $h : \mathbb{N}^p \longrightarrow \mathbb{N}$, et $g_1, \dots, g_p : \mathbb{N}^n \longrightarrow \mathbb{N}$ sont récursives primitives, alors $f : \mathbb{N}^n \longrightarrow \mathbb{N}$ définie par $f(x_1, \dots, x_n) = h(g_1(x_1, \dots, x_n), \dots, g_p(x_1, \dots, x_n))$ est récursive primitive.

iii. et qui est clos par le *schéma de récurrence primitive* :
si $g : \mathbb{N}^p \longrightarrow \mathbb{N}$ et $h : \mathbb{N}^{p+2} \longrightarrow \mathbb{N}$ sont récursives primitives, alors $f : \mathbb{N}^{p+1} \longrightarrow \mathbb{N}$ est récursive primitive, f définie par :

$$\begin{aligned} &f(a_1, \dots, a_p, 0) = g(a_1, \dots, a_p) \\ &f(a_1, \dots, a_p, x+1) = h(a_1, \dots, a_p, x, f(a_1, \dots, a_p, x)) \,. \end{aligned}$$

Un prédicat P sur N^p, resp. un sous-ensemble E de N^p, est un *prédicat récursif primitif*, resp. *un sous-ensemble récursif primitif*, quand sa fonction caractéristique est récursive primitive.

Remarque : On peut prendre $p = 0$ dans le schéma de récurrence récursive primitive :
si $b \in \mathbb{N}$, si $h : \mathbb{N}^2 \longrightarrow \mathbb{N}$ est récursive primitive, alors $f : \mathbb{N} \longrightarrow \mathbb{N}$ est récursive primitive, f définie par :

$$\begin{aligned} &f(0) = b \\ &f(x+1) = h(x, f(x)) \,. \end{aligned}$$

En effet il suffit d'ajouter un argument inutile : on définit une fonction auxiliaire $f' : \mathbb{N}^2 \to \mathbb{N}$ par récurrence primitive à partir de la fonction g constante égale à b, et de la fonction $h'(a, x, f(x)) = h({p^2}_3(a, x, f(x)), {p^3}_3(a, x, f(x)))$. On a $f(x) = f'(x, x)$.

Exercice 1. Montrer que l'ensemble des fonctions récursives primitives est un ensemble dénombrable.

Exercice 2 (exemples, cas particuliers du schéma de récurrence récursive primitive).

1. Montrer que les fonctions constantes de $\mathbb{N}^k \to \mathbb{N}$, $k \in \mathbb{N}$, sont récursives primitives.

2. Montrer que les trois fonctions de $\mathbb{N}$ dans $\mathbb{N}$, $f : x \mapsto x + 2$, $g : x \mapsto 2x$ et $h : x \mapsto 2x + 1$ sont récursives primitives.

3. Montrer que l'addition, la multiplication, et l'exponentielle sont des fonctions récursives primitives.

4. Montrer que la fonction sg de $\mathbb{N}$ dans $\mathbb{N}$, qui à 0 associe 0 et qui à tous les autres entiers associe 1 ainsi que la fonction $\overline{sg}$ de $\mathbb{N}$ dans $\mathbb{N}$, qui à 0 associe 1 et qui à tous les autres entiers associe 0 sont récursives primitives (se servir de la récurrence primitive).

5. Montrer que l'ensemble des fonctions récursives primitives est clos sous le schéma de *définition par itération*, qui a une fonction g de $\mathbb{N}^p$ dans $\mathbb{N}$ et à une fonction h de $\mathbb{N}^{p+1}$ dans $\mathbb{N}$ associe la fonction f de $\mathbb{N}^{p+1}$ dans $\mathbb{N}$ définie par :

$$\begin{aligned} &f(a_1, \dots, a_p, 0) = g(a_1, \dots, a_p) \\ &f(a_1, \dots, a_p, x+1) = h(a_1, \dots, a_p, f(a_1, \dots, a_p, x))) \,. \end{aligned}$$

Montrer ensuite que les fonctions introduites jusqu'à présent dans cet exercice se définissent à partir des fonctions de base et du schéma d'itération (on verra des exemples de fonctions qui utilisent naturellement pour leur définition un schéma récursif primitif qui n'est pas un schéma d'itération : le prédécesseur de l'exercice 4, la factorielle, ...).

6. Montrer que l'ensemble des fonctions récursives primitives est clos *par définition par cas* sur un prédicat récursif primitif : si g et h sont des fonctions récursives primitives de $\mathbb{N}^p$ dans $\mathbb{N}$, et P un prédicat récursif primitif sur N^p, alors la fonction f de $\mathbb{N}^p$ dans $\mathbb{N}$ définie ci-dessous est récursive primitive :

$$\text{si } P(a_1, \dots, a_p) \text{ alors } f(a_1, \dots, a_p) = g(a_1, \dots, a_p) \text{ sinon } f(a_1, \dots, a_p) = h(a_1, \dots, a_p) \,.$$

Exercice 3 (somme et produit bornés). Montrer que si $f : \mathbb{N}^{p+1} \longrightarrow \mathbb{N}$ est récursive primitive, les fonctions g somme bornée de f, et h produit borné de f, toutes deux de $\mathbb{N}^{p+1} \longrightarrow \mathbb{N}$ définies par

$$g(a_1, \dots, a_p, x) = \sum_{i=0}^{x} f(a_1, \dots, a_p, i) \quad h(a_1, \dots, a_p, x) = \prod_{i=0}^{x} f(a_1, \dots, a_p, i)$$

sont récursives primitives.

Exercice 4 (prédécesseur, comparaison).

1. Montrer que la fonction $pred : \mathbb{N} \longrightarrow \mathbb{N}$ qui vaut 0 en 0 et $n-1$ en $n > 0$ est récursive primitive.
2. Montrer que la fonction $\dot{-}: \mathbb{N}^2 \longrightarrow \mathbb{N}$ définie par « $x \dot{-} y = x - y$ si $x \geq y$, $x \dot{-} y = 0$ sinon », ainsi que la fonction : $x, y \mapsto |x - y|$ sont récursives primitives.
3. Montrer que les prédicats de comparaison $\leq, \geq, <, >, =, \neq$ sont récursifs primitifs.

Exercice 5 (Prédicats récursifs primitifs, opérations booléennes).

1. Montrer que l'ensemble des prédicats récursifs primitifs d'arité quelconque est clos sous les opérations booléennes (conjonction, disjonction, négation). Par exemple on montrera que si les prédicats $P[x_1, \ldots, x_p, y_1, \ldots, y_n]$ et $Q[x'_1, \ldots, x'_q, y_1, \ldots, y_n]$ sont récursifs primitifs, alors le prédicat
$$P[x_1, \ldots, x_p, y_1, \ldots, y_n] \wedge Q[x'_1, \ldots, x'_q, y_1, \ldots, y_n]$$
est récursif primitif.
2. En déduire que l'ensemble des ensembles primitifs récursifs est clos par réunion interscetion et passage au complémentaire.

Exercice 6. Montrer que les sous-ensembles finis et cofinis des $\mathbb{N}^p$, $p \in \mathbb{N}$, sont récursifs primitifs.

Exercice 7 (minimisation bornée). On propose dans cet exercice une démonstration de la clôture de l'ensemble des fonctions récursives primitives sous le schéma de *minimisation bornée*, qui à un prédicat primitif récursif B sur $\mathbb{N}^{p+1}$ associe la fonction f de $\mathbb{N}^{p+1}$ dans $\mathbb{N}$ définie par :

$$\begin{array}{rcll} f(a_1, \ldots, a_p, x) & = & \text{le plus petit entier } t \leq x \text{ tel que } B(a_1, \ldots, a_p, t) & \text{s'il existe un tel entier,} \\ f(a_1, \ldots, a_p, x) & = & 0 & \text{s'il n'existe pas de tel entier.} \end{array}$$

On note $f(a_1, \ldots, a_p, x) = \mu t \leq x.B(a_1, \ldots, a_p, t)$.

1. Soit un prédicat récursif primitif B sur $\mathbb{N}^{p+1}$, montrer que la fonction $b : \mathbb{N}^{p+1} \longrightarrow \mathbb{N}$ est récursive primitive, où b est définie par :
$$\begin{array}{rcll} b(a_1, \ldots, a_p, x) & = & 0 & \text{s'il existe un entier } t \leq x \text{ tel que } B(a_1, \ldots, a_p, t), \\ b(a_1, \ldots, a_p, x) & = & 1 & \text{s'il n'existe pas de tel entier} \end{array}$$
(se servir de l'exercice 3).
2. En déduire que l'ensemble des fonctions récursives primitives est clos sous le schéma de minimisation borné.

Exercice 8 (quantifications bornées). Montrer que l'ensemble des prédicats récursifs primitifs est clos par quantification existentielle et universelle bornée (on peut utiliser la première question de l'exercice 7 ou directement l'exercice 3).

Exercice 9 (division euclidienne). Montrer que les fonctions $q : \mathbb{N}^2 \longrightarrow \mathbb{N}$ et $r : \mathbb{N}^2 \longrightarrow \mathbb{N}$, où $q(n,p)$ est le quotient et $r(n,p)$ le reste de la division de n par p sont des fonctions récursives primitives. En déduire que le prédicat binaire $|$ ($a \mid b$ signifie a est un diviseur de b) est récursif primitif.

Exercice 10 (nombres premiers). Soit $p : \mathbb{N} \longrightarrow \mathbb{N}$ la fonction telle que $p(n)$ soit le $n+1$-ième nombre premier.

1. Montrer que le prédicat «être premier» est récursif primitif.
2. Montrer que $p(n+1) \leq p(n)! + 1$ et que la fonction factorielle est récursive primitive.
3. Montrer que la fonction p est récursive primitive.

Exercice 11 (codage des couples et k-uplets). Soient α_2 la bijection de Cantor $\mathbb{N} \times \mathbb{N}$ dans $\mathbb{N}$, définie par :

$$\alpha_2(n,p) = (\sum_{i=0}^{n+p} i) + p = \frac{(n+p+1)(n+p)}{2} + p$$

(faire un dessin).

1. Vérifier que α_2 est bien bijective et récursive primitive. Vérifier que α_2 est croissante sur chacune de ses deux composantes.
2. Définir de façon récursive primitive les deux «projections» associées π_2^1 et π_2^2 vérifiant :
$$\alpha_2(\pi_2^1(c), \pi_2^2(c)) = c \quad \pi_2^1(\alpha_2(n,p)) = n \quad \pi_2^2(\alpha_2(n,p)) = p\ .$$

3. On définit par récurrence sur $k \leq 1$ les fonctions $\alpha_k : \mathbb{N}^p \to \mathbb{N}$ par :

$$\alpha_1(n) = n \quad \alpha_{k+1}(n_1, \ldots, n_{k+1}) = \alpha_2(n_1, \alpha_k(n_2, \ldots, n_{k+1})) \ .$$

Montrer que, pour tout $k \geq 1$, α_k est une bijection primitive récursive de $\mathbb{N}^k \to \mathbb{N}$ et définir de façon récursive primitive les « projections » $\pi_k^i : \mathbb{N} \to \mathbb{N}$ associées. Vérifier que α_k est croissante sur chacune de ses composantes. On écrira aussi $\langle x_1, \ldots, x_k \rangle$ pour $\alpha_k(x_1, \ldots, x_k)$.

Exercice 12 (Définitions par récurrences mutuelles). Utiliser la fonction α_k pour montrer que si les fonctions $g_1, \ldots, g_k : \mathbb{N}^n \to \mathbb{N}$, et $h_1, \ldots, h_k : \mathbb{N}^{n+k+1} \to \mathbb{N}$ sont primitives récursives, alors, les fonctions $f_1, \ldots, f_k$ définies ci-dessous sont primitives récursives (on écrit $\vec{a}$ pour $a_1, \ldots, a_n$).

$$\begin{array}{lcl} f_1(\vec{a}, 0) = g_1(\vec{a}) & & f_k(\vec{a}, 0) = g_k(\vec{a}) \\ f_1(\vec{a}, x+1) = h_1(\vec{a}, x, f_1(\vec{a}, x), \ldots, f_k(\vec{a}, x)) & \cdots & f_k(\vec{a}, x+1) = h_k(\vec{a}, x, f_1(\vec{a}, x), \ldots, f_k(\vec{a}, x)) \end{array}$$

Exercice 13 (Un codage bijectif des suites finies). On obtient la fonction :: (notation infixe) en translatant α_2 de 1 :

$$x :: y = 1 + \alpha_2(x, y)$$

On obtient ainsi une fonction récursive primitive bijective de $\mathbb{N}^2 \to \mathbb{N}^*$. On appelle hd et tl les fonctions vérifiant :

$$\begin{array}{ll} hd(0) = 0 & tl(0) = 0 \\ hd(x :: y) = x & tl(x :: y) = y \end{array}$$

On définit une fonction *liste* de l'ensemble $\mathcal{S}$ des suites finies d'entiers dans $\mathbb{N}$ de la façon suivante (on note $[a_0; \ldots; a_n] =_d \mathit{liste}((a_0, \ldots, a_n)))$:

$$\begin{array}{l} [\,] = 0 \\ [a_0; \ldots; a_n] = a_0 :: [a_1; \ldots; a_n] \end{array}$$

Montrer que la fonction *liste* est bijective, et que les fonctions hd et tl sont récursives primitives.

Exercice 14 (récurrence sur la suite des valeurs). On va utiliser les entiers pour coder des ensembles définis par induction : termes, formules ... Pour cela nous aurons besoin d'une définition de fonction par récurrence qui fait appel, pour définir la fonction en n à un ou plusieurs entiers strictement plus petits que n, et non seulement au prédécesseur comme le schéma de récurrence primitive. C'est le schéma de définition par *récurrence primitive sur la suite des valeurs.*

1. Démontrer que l'ensemble des fonctions récursives primitives est clos par le schéma de récurrence sur la suite des valeurs suivant : si $g : \mathbb{N}^p \longrightarrow \mathbb{N}$ et $h : \mathbb{N}^{p+2} \longrightarrow \mathbb{N}$ sont récursives primitives, alors $f : \mathbb{N}^{p+1} \longrightarrow \mathbb{N}$ est récursive primitive, f définie par :

$$\begin{array}{l} f(a_1, \ldots, a_p, 0) = g(a_1, \ldots, a_p) \\ f(a_1, \ldots, a_p, x+1) = h\left(a_1, \ldots, a_p, x, [f(a_1, \ldots, a_p, x); \ldots; f(a_1, \ldots, a_p, 0)]\right) \ . \end{array}$$

2. Montrer que t la fonction *nthl* qui à l et i associe la suite codée par l à partie du $i+1$-ième élément (0 sinon), et la fonction *nth* qui à l et i associe le i-ème élément de la suite codée par l (0 sinon), sont récursives primitives.

3. Montrer que si $g : \mathbb{N}^p \longrightarrow \mathbb{N}$, $h : \mathbb{N}^{p+k+1} \longrightarrow \mathbb{N}$ sont récursives primitives, et si $p_1, \ldots, p_k : \mathbb{N} \longrightarrow \mathbb{N}$ sont des fonctions récursives primitives vérifiant chacune :

$$\forall x \in \mathbb{N} \; p_i(x) \leq x$$

alors $f : \mathbb{N}^{p+1} \longrightarrow \mathbb{N}$, définie par :

$$\begin{array}{l} f(a_1, \ldots, a_p, 0) = g(a_1, \ldots, a_p) \\ f(a_1, \ldots, a_p, x+1) = h\left(a_1, \ldots, a_p, x, f(a_1, \ldots, a_p, p_1(x)), \ldots, f(a_1, \ldots, a_p, p_k(x))\right) \end{array}$$

est récursive primitive. Ce schéma est indépendant du codage des suites. Mais il peut-être vu comme un cas particulier de la récurrence sur la suite des valeurs : la fonction au rang x peut dépendre seulement d'un nombre fixe des valeurs de la fonction en les y avant x, et non de toutes ces valeurs.

Exercice 15 (récurrence sur les listes). **1.** Montrer que l'ensemble des fonctions récursives primitives est clos par le schéma de récurrence sur les listes (préciser pourquoi f est bien définie) : si $g : \mathbb{N}^p \longrightarrow \mathbb{N}$ et $h : \mathbb{N}^{p+3} \longrightarrow \mathbb{N}$ sont récursives primitives, alors $f : \mathbb{N}^{p+1} \longrightarrow \mathbb{N}$ est récursive primitive, f définie par

$$\begin{array}{l} f(a_1, \ldots, a_p, [\,]) = g(a_1, \ldots, a_p) \\ f(a_1, \ldots, a_p, x :: l) = h\left(a_1, \ldots, a_p, x, l, f(a_1, \ldots, a_p, l)\right) \ . \end{array}$$

et l'utiliser pour montrer que la fonction *mem* fonction caractéristique de l'appartenance d'un entier n à une suite codée par l, la fonction @ vérifiant que $l \,@\, l'$ est le code de la concaténation des suites codées par l et l', la fonction *length* qui à un entier l associe la longueur de la suite codée par l, sont récursives primitives.

2. Montrer que si $f : \mathbb{N}^{p+1} \to \mathbb{N}$ est récursive primitive, alors la fonction map_f qui à l codant $(u_i)_{i\leq n}$ associe $map_f(l)$ codant $(f(a_1,\ldots,a_p,u_i))_{i\leq n}$ est récursive primitive.

Montrer que la fonction *concat* qui à un entier l codant une suite de suites $\left((u_i)_{i\leq n_i}\right)_{i\leq p}$ associe l'entier $concat(l)$ codant la suite des entiers de chaque suite $(u_i)_{i\leq n_i}$ dans le même ordre est récursive primitive.
Montrer que la fonction *subst* qui à trois entiers l, k, v, associe le code la suite obtenue en remplaçant dans la suite codée par l toutes les occurrences de v par les entiers de la suite codée par k est récursive primitive (on peut se servir des deux fonctions précédentes).

On peut utiliser la décomposition en nombre premiers pour un autre codage (non bijectif) des suites (voir feuille sur les fonctions élémentaires).

Exercice 16 (récurrence avec substitution de paramètre). On appelle *schéma de récurrence avec substitution de paramètre* le schéma de récurrence suivant (énoncé ici avec un seul paramètre) qui étant données $g,\ \gamma : \mathbb{N} \longrightarrow \mathbb{N}$ et $h : \mathbb{N}^3 \longrightarrow \mathbb{N}$ définit $f : \mathbb{N}^2 \longrightarrow \mathbb{N}$.

$$\begin{array}{l} f(a,0) = g(a) \\ f(a,x+1) = h(a,x,f(\gamma(a),x))\,. \end{array}$$

Intuitivement, ce schéma conserve le fait d'être calculable (le calcul termine puisque la variable de récurrence décroit). On veut montrer que l'ensemble des fonctions primitives récursives est clos sous ce schéma. On suppose dans la suite g, γ et h primitives récursives, et f définie comme ci-dessus. On note $\gamma^p(x) = \underbrace{\gamma\circ\cdots\circ\gamma}_{p}(x)$ ($\gamma^0 = \lambda x.x$).

1. Montrer que la fonction F définie ci-dessous est primitive récursive

$$\begin{array}{l} F(p,a,0) = g\left(\gamma^p(a)\right) \\ F(p,a,x+1) = h\left(\gamma^{p\dot{-}(x+1)}(a),x,F(p,a,x)\right)\,. \end{array}$$

2. Montrer que :

$$\forall x,a,p\in\mathbb{N}\ \left(x\leq p\to F(p,a,x) = f(\gamma^{p\dot{-}x}(a),x)\right)$$

et en déduire que f est primitive récursive.

3. Application : montrer que la fonction $inc : \mathbb{N}^2 \to \mathbb{N}$ qui à i et $l = [a_0;\ldots;a_i;\ldots;a_n]$ associe $inc(i,l) = [a_0;\ldots;a_i+1;\ldots;a_n]$ quand $i\leq n$, $inc(i,l) = l$ sinon, est primitive récursive.

Exercice 17 (récurrence double sans imbrication). On sait que la récurrence double ne conserve pas en général le fait d'être primitif récursif (la fonction d'Ackermann est définie par récurrence double). On peut montrer que s'il n'y a pas *imbrication* des appels récursifs dans la récurrence double, alors celle-ci reste "primitive récursive".

On suppose que a et b sont des entiers, que h est une fonction primitive récursive à 4 arguments, que h_2 est une fonction primitive récursive à 2 arguments.

Le schémas qui suivent ne comportent pas de paramètres, ce qui simplifie les notations, les démonstrations étant essentiellement les mêmes en présence de paramètres.

1. Montrer que la fonction f définie par :

$$\begin{array}{rcl} f(0,y) &=& a\,, \\ f(x+1,0) &=& b\,, \\ f(x+1,y+1) &=& h(x,y,f(x,y),f(x+1,y))\,. \end{array}$$

est primitive récursive (on peut utiliser le codage des couples et la "récurrence sur la suite des valeurs").

2. (généralisation, difficile) Montrer que la fonction f définie par :

$$\begin{array}{rcl} f(0,y) &=& a\,, \\ f(x+1,0) &=& b\,, \\ f(x+1,y+1) &=& h(x,y,f(x,h_2(x,y)),f(x+1,y))\,. \end{array}$$

est primitive récursive. On peut procéder comme à la question précédente, mais en cherchant pour h_2 donné une nouvelle fonction α' de codage des couples (injective mais non nécessairement bijective) vérifiant :

$$\alpha'(x+1,y) < \alpha'(x+1,y+1) \text{ et } \alpha'(x,h_2(x,y)) < \alpha'(x+1,y+1)\,.$$