Université de Paris **(Date: 23/09/2020)**
UFR Mathématiques

Calculabilité et incompletude TD1

Professor: P. Rozière Juan Ignacio Padilla*,* M2 LMFI

Exercice 1. Montrer que l'ensemble des fonctions primitives recursives est un ensemble dénombrable.
Solution: On peut définir par récurrence

$$\begin{aligned}\mathcal{F}_0 &= \{\lambda x.0, \lambda x.s(x)\} \cup \{p_k^i, 1 \leq i \leq k\}_{k\in\mathbb{N}} \\ \mathcal{F}_{n+1} &= \{f \in \mathbb{N}^{\mathbb{N}^k}, \exists g, h \in \mathcal{F}_n, f = \mathrm{Rec}(g,h)\}_{n\in\mathbb{N}} \\ &\cup \{f \in \mathbb{N}^{\mathbb{N}^k}, \exists g_1, \dots, g_m, h \in \mathcal{F}_n, f \equiv h(g_1, \dots, g_m)\}_{k\in\mathbb{N}}\end{aligned}$$

Chaque un des $\mathcal{F}_n$ est dénombrable puisque les opérations Rec et la composition n'exigent que finis arguments. On a que l'ensembre des fonctions primitives recursives est $\mathcal{F} = \bigcup_n \mathcal{F}_n$, et par conséquent il est dénombrable.

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

(1) Montrer que les fonctions constantes sont récursives primitives. Par récurrence: la fonction $\lambda x.1$ est égale a la composition entre $s(x)$ et la fonction nulle. Maintenant, si $f(x) = \lambda x.k$ est récursive primitive, donc $\lambda x.k + 1 = s(f(x))$, qui est récursive primitive par schéma de composition.

(2) Montrer que $x \mapsto x+2$, $x \mapsto 2x$ et $x \mapsto 2x+1$ sont récursives primitives. $f(x) = \lambda x.x+2 = s(s(p_1^1(x)))$, la fonction de duplication est définie par récurrence primitive comme $g(0) = 0$ et $g(x+1) = f(p_2^1(x, g(x))$. Enfin $h(x) = s(g(x))$.

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

$$\begin{aligned}+(x,0) &= x = p_1^1(x) \\ +(x,y+1) &= s(p_3^3(x,y,+(x,y))) \\ \times(x,0) &= 0 \\ \times(x,y+1) &= +(p_3^3(x,y,\times(x,y)), p_3^1(x,y,\times(x,y)) \\ \exp(x,0) &= 1 \\ \exp(x,y+1) &= \times(p_3^3(x,y,\exp(x,y)), p_3^1(x,y,\exp(x,y))\end{aligned}$$

(4) Montrer que la fonction sg qui a 0 associe 0 et qui à tous les autres entiers associe 1 ainsi que la fonction $\bar{\mathrm{sg}}$ qui a 0 associe 1 et qui à tous les autres entiers associe 0 sont récursives primitives.

$$\begin{aligned}\mathrm{sg}(0) &= 0 \\ \mathrm{sg}(x+1) &= \lambda xy.1(x, \mathrm{sg}(0))\end{aligned}$$

L'autre cas est le même.

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

$f : \mathbb{N}^p \to \mathbb{N}$ définie par:

$$f(a_1, \ldots, a_p, 0) = g(a_1, \ldots, a_p)$$
$$f(a_1, \ldots, a_p, x+1) = h(a_1, \ldots, a_p, f(a_1, \ldots, a_p, x))).$$

On peut posser

$$f(a_1, \ldots, a_p, 0) = g(a_1, \ldots, a_p)$$
$$f(a_1, \ldots, a_p, x+1) = h(p^1_{p+2}(\bar{a}, x, f(\bar{a}, x)), \ldots, p^p_{p+2}(\bar{a}, x, f(\bar{a}, x)), p^{p+2}_{p+2}(\bar{a}, x, f(\bar{a}, x)))$$

pour exprimer f sous forme récursive primitive. 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 a

$$\begin{aligned} +(x, 0) &= x \\ +(x, y+1) &= s(x, +(x, y)) \\ \times(x, 0) &= x \\ \times(x, y+1) &= +(x, \times(x, y)) \\ \exp(x, 0) &= x \\ \exp(x, y+1) &= \times(x, \exp(x, y)) \end{aligned}$$

(6) Montrer que l'ensembre 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 $\mathbb{N}^p$, alors la fonction f de $\mathbb{N}^p$ dans $\mathbb{N}$ définie ci-dessous est récursive primitive:

$$f(a_1, \ldots, a_p) = \begin{cases} g(a_1, \ldots, a_p) & \text{si } P(a_1 \ldots, a_n) \\ h(a_1, \ldots, a_n) & \text{sinon} \end{cases}$$

On a $f(\bar{a}) = g(\bar{a})\chi_P(\bar{a}) + h(\bar{a})\chi_P(\bar{a})$.

Exercice 3 (somme et produit bornés). Montrer que si $f : \mathbb{N}^{p+1} \to \mathbb{N}$ est récursive primitive, les fonctions g et h définites par

$$g(\bar{a}, x) = \sum_{i=0}^{x} f(\bar{a}, i) \text{ et } h(\bar{a}, x) = \prod_{i=0}^{x} f(\bar{a}, i)$$

sont récursives primitives.

Solution: On a

$$\begin{aligned} g(\bar{a}, 0) &= f(\bar{a}, 0) \\ g(\bar{a}, x+1) &= g(\bar{a}, x) + f(\bar{a}, x+1) \\ h(\bar{a}, 0) &= f(\bar{a}, 0) \\ h(\bar{a}, x+1) &= h(\bar{a}, x) \times f(\bar{a}, x+1) \end{aligned}$$

Exercice 4 (prédecesseur, comparaison)

(1) Montrer que la fonction pred $\mathbb{N} \to \mathbb{N}$ qui vaut 0 en 0 et $n-1$ en $n > 0$ est récursive primitive.

$$\begin{aligned} \text{pred}(0) &= 0 \\ \text{pred}(n+1) &= n \end{aligned}$$

(2) Montrer que $x \dot{-} y = x - y$ si $x \geq y$ et 0 sinon, ainsi que la fonction $x, y \mapsto |x - y|$ sont récursives primitives.

$$x \dot{-} 0 = x$$
$$x \dot{-} (y+1) = \mathrm{pred}(x \dot{-} y)$$

(3) Montrer que les prédicats de comparaison $\leq, \geq, <, <, =, \neq$ sont récursifs primitifs. On a $\chi_{\leq}(x,y) = \overline{\mathrm{sg}}(x \dot{-} y)$, $\chi_{\geq}(x,y) = \overline{\mathrm{sg}}(y \dot{-} x)$, $\chi_{=}(x,y) = \chi_{\leq}(x,y)\chi_{\geq}(x,y)$, $\chi_{\neq}(x,y) = \overline{\mathrm{sg}}(\chi_{=}(x,y))$, $\chi_{<}(x,y) = \chi_{\leq}(x,y)\chi_{\neq}(x,y)$, $\chi_{>}(x,y) = \chi_{\geq}(x,y)\chi_{\neq}(x,y)$.

Exercice 5 (Prédicats récursifs primitifs, opérations booléennes

(1) Montrer que l'ensemble des prédicats récursifs primitifs d'arité quelconque est clos sous les opérations booléenenes.
(2) En déduire que l'ensemble des ensembles récursifs primitifs est clos par réunion, intersection et passage au complémentaire.

Solution: (1) et (2) Si $P[\bar{x}, \bar{y}]$ et $Q[\bar{x}', \bar{y}]$ sont prédicats récursifs primitifs, on a

$$\chi_{P\wedge Q}(\bar{x}, \bar{x}', \bar{y}) = \chi_P(\bar{x}, \bar{y})\chi_Q(\bar{x}', \bar{y})$$
$$\chi_{P\vee Q}(\bar{x}, \bar{x}', \bar{y}) = \mathrm{sg}(\chi_P(\bar{x}, \bar{y}) + \chi_Q(\bar{x}', \bar{y}))$$
$$\chi_{\neg P}(\bar{x}, \bar{y}) = \overline{\mathrm{sg}}(\chi_P(\bar{x}, \bar{y}))$$

Même pour ensembles recursifs primitifs dans $\mathbb{N}^p$.

Exercice 6. Montrer que les sous-ensembles finis et cofinis des $\mathbb{N}^p$ sont récursifs primitifs.
Solution: Si $p = 0$, $\varnothing$ a comme fonction characteristique la fonction nulle. Si $p > 0$ et $A \subseteq \mathbb{N}^p$ est fini, on a $A = \{\bar{a}_1, \ldots, \bar{a}_n\}$. On peut posser

$$\chi_A(\bar{x}) = \begin{cases} 1 & \text{si } \bigvee_{i=0}^{n} \bar{x} = \bar{a}_i \\ 0 & \text{sinon} \end{cases}$$

Par *définition par cas*, A est récursif primitif. Notez que le prédicat $P[\bar{x}] : \bar{x} = \bar{a}$ a comme fonction caractéristique $\chi_P(\bar{x}) = \chi_{=}(\bar{x}, \bar{a})$, il est donc primitif récursif. Si A est cofini, on a $\chi_A(\bar{x}) = \overline{\mathrm{sg}}(\chi_{\mathbb{N}^p \setminus A}(\bar{x}))$.

Exercice 7 (minimisation bornée) Le schéma de *minimisation bornée* associe a un prédicat récursif primitif $B \subseteq \mathbb{N}^{p+1}$ la fonction $f : \mathbb{N}^{p+1} \to \mathbb{N}$ définie par:

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

On note $f(\bar{a}, x) = \mu t \leq x B(\bar{a}, t)$.

(1) Soit un prédicat récursif primitif $B \subseteq \mathbb{N}^{p+1}$, montrer que la fonction $b : \mathbb{N}^{p+1} \to \mathbb{N}$ est récursive primitive, oú b est définie par:

$$b(a_1, \ldots, a_p, x) = 0 \text{ s'il existe un entier } t \leq x \text{ tel que } B(\bar{a}, t)$$
$$b(a_1, \ldots, a_p, x) = 1 \text{ s'il n'existe pas de tel entier}$$

On peut posser

$$b(\bar{a}, x) = \overline{\mathrm{sg}}\left(\sum_{t=0}^{x} \chi_B(\bar{a}, t)\right)$$

(2) En déduire que l'ensemble des fonctions récursives primitives est clos sous le schéma de minimisation borné. On peut posser, en utilisant l'aide de la fonction b,

$$f(\bar{a}, x) = \sum_{t=0}^{x} b(\bar{a}, x).$$

Exercice 8 (quantifications bornées). Montrer que l'ensemble des prédicats récursifs primitifs est clos par quantification existentielle et universelle bornée.
Solution: Si P est un prédicat primitif récursif, et on défini

$$P_e[\bar{x}, y] = \exists z \leq y P[\bar{x}, z]$$
$$P_q[\bar{x}, y] = \forall z \leq y P[\bar{x}, z]$$

Alors,

$$\chi_{P_e}(\bar{x}, y) = \operatorname{sg}\left(\sum_{t=0}^{y} \chi_P(\bar{x}, t)\right)$$
$$\chi_{P_q}(\bar{x}, y) = \left(\prod_{t=0}^{y} \chi_P(\bar{x}, t)\right)$$

Exercice 9 (division euclidienne). Montrer que les fonctions $q : \mathcal{N}^2 \to \mathbb{N}$ et $r : \mathbb{N}^2 \to \mathbb{N}$ 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 déduite que le prédicat binaire $a|b$ est récursif primitif.
Solution:

$$q(n, p) = \mu t \leq n(pt \leq n \wedge p(t + 1) > n)$$
$$r(n, p) = n \dot{-} (p \times q(n, p))$$

Et on a que

$$\chi_{n|p}(n, p) = \begin{cases} 1 & \text{si } r(n, p) = 0 \\ 0 & \text{sinon} \end{cases}$$

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

(1) Montrer que le prédicat «être premier» est récursif primitif.

$$p \text{ est premier ssi } p > 1 \wedge \forall x \leq p(\neg x|p \vee x = 1 \vee x = p)$$

(2) Montrer que $p(n + 1) \leq p(n)! + 1$ et que la fonction factorielle est récursive primitive.
Soit q premier tel que $q|p(n)! + 1$, on sait que $q \notin \{p(0), \ldots, p(n)\}$ (le cas contraire impliquerait l'absurd $q|1$), ceci implique que $p(n + 1) \leq q \leq p(n)! + 1$. On a aussi

$$0! = 1$$
$$(n + 1)! = (n + 1)n!$$

Ce que montre que $n!$ est primitif récursif.

(3) Montrer que la fonction p est récursive primitive.
On considére la fonction primitive récursive

$$p'(n, y_1, y_2) = \mu t \leq y_1(t \text{ est premier} \wedge y_2 \leq t).$$

Alors,

$$p(0) = 2$$
$$p(n+1) = p'(n, p(n)! + 1, p(n))$$

Ce que montre que $p(n)$ est primitive récursive.

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

$$\alpha(n,p) = \left(\sum_{i=0}^{n+p} i\right) + p.$$

(1) Verifier que α est bien bijective et récursive primitive. Vérifier que α est croissante sur chacune de ses deux composantes. Si $m = n + n'$ on a, par tout p

$$\alpha(m,p) = \left(\sum_{i=0}^{m+p} i\right) + p = \left(\sum_{i=0}^{n+p} i\right) + \left(\sum_{i=n+p+1}^{n+n'+p} i\right) + p \geq \left(\sum_{i=0}^{n+p} i\right) + p = \alpha(n,p).$$

Si $p \leq q$ c'est evident que par tout n

$$\left(\sum_{i=0}^{n+p} i\right) + p \leq \left(\sum_{i=0}^{n+q} i\right) + q.$$

À partir de l'ex 3. il est claire que α est récursive primitive. On va montrer injectivité, denote $(\sum_{i=0}^{n} i) = \Delta(n)$
Soit $(n,p) \neq (m,q)$, si $n + p = m + q$ donc $\Delta(n+p) = \Delta(m+q)$, et si on suppose que $\alpha(n,p) = \alpha(m,q)$ cela impliquerait que $p = q$ et pourtant $n = m$, une contradiction. Pour la surjectivité, soit $m \in \mathbb{N}$, prends le plus petit x tel que $\Delta(x) \leq m \leq \Delta(x+1)$ et prends $r = m - x$. Note que $r \leq x$, sinon on aurait $r > x \Rightarrow m = \Delta(x) + r \geq \Delta(x+1)$ ce qui contredit la minimalité de x. Alors, $x = r + m$ et $m = \Delta(r+m) + r = \alpha(m,r)$

(2) Définit de façon récursive primitive les deux projections asosciés π_2^1 et π_2^2 vérifiant

$$\alpha(\pi_2^1(c), \pi_2^2(c)) = c\ ,\ \pi_2^1(\alpha(n,p)) = n\ ,\ \pi_2^2(\alpha(n,p)) = p.$$

Il est evident que $n, p \leq \alpha(n,p)$, on peut poser

$$\pi_2^1(c) = (\mu z \leq c)(\exists t \leq c)(\alpha(z,t) = c)$$
$$\pi_2^2(c) = (\mu z \leq c)(\exists t \leq c)(\alpha(t,z) = c)$$

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

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

Montrer que, pour tout $k \leq 1$, α_k est une bijection récursive primitive et définir de façon récursive 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 $\alpha_k(x_1, \ldots, x_k)$.
Le fait que α_k est bijective et primitive récursive est montré facilement par récurrence, parce que α_k est composition des fonctions primitives récursives. Par récurrence, si π_k^i sont définies par $i = 1, \ldots, k$, on définit

$$\pi_{k+1}^1(n_1, \ldots, n_{k+1}) = \pi_2^1(\alpha(n_1, \alpha_k(n_2, \ldots, n_{k+1})))$$
$$\pi_{k+1}^i(n_1, \ldots, n_{k+1}) = \pi_k^{i-1}(\pi_2^2(\alpha(n_1, \alpha_k(n_2, \ldots, n_{k+1}))))\ \text{ par } i \in \{2, \ldots, k+1\}$$

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

$$\begin{aligned}
&f_1(\bar{a}, 0) = g_1(\bar{a})\\
&\vdots\\
&f_k(\bar{a}, 0) = g_k(\bar{a})\\
&f_1(\bar{a}, x+1) = h_1(\bar{a}, x, f_1(\bar{a}, x), \dots, f_k(\bar{a}, x))\\
&\vdots\\
&f_k(\bar{a}, x+1) = h_k(\bar{a}, x, f_1(\bar{a}, x), \dots, f_k(\bar{a}, x))
\end{aligned}$$

On peut poser simplement par $i \leq i \leq k$

$$\begin{aligned}
f_i(\bar{a}, 0) &= \pi_k^i(\alpha_k(g_1(\bar{a}), \dots, g_k(\bar{a}))\\
f_i(\bar{a}, x+1) &= \pi_k^i\Big(\alpha_k\Big(h_1(\bar{a}, x, f_1(\bar{a}, x), \dots, f_k(\bar{a}, x)), \dots, h_k(\bar{a}, x, f_1(\bar{a}, x), \dots, f_k(\bar{a}, x)\Big)\Big)
\end{aligned}$$

Par schéma de composition, f_i est primitive récursive.

Exercice 13 (Un codage bijectif des suites finies). On obtient la fonction ::

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

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

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

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

$$\begin{aligned}
[\ ] &= 0\\
[a_0; \dots; a_n] &= a_0 :: [a_1; \dots; a_n]
\end{aligned}$$

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

$$\begin{aligned}
\mathrm{hd}(c) &= \pi_2^1(c \dot{-} 1)\\
\mathrm{tl}(c) &= \pi_2^2(c \dot{-} 1)
\end{aligned}$$

Par obtenir que liste est injective, soient $[a_0; \dots; a_n) = [b_0; \dots; b_{n+k}]$ pour certain $k \geq 0$, donc

$$\begin{aligned}
&a_0 :: [a_1; \dots; a_n] = b_0 :: [b_1; \dots; b_{n+k}]\\
\Rightarrow &a_0 = b_0 \wedge [a_1; \dots; a_n] = [b_1; \dots; b_{n+k}]
\end{aligned}$$

On peut repeter répéter cet argument commençant par $[a_1; \dots; a_n] = [b_1; \dots; b_{n+k}]$ et arriver à

$$\bigwedge_{i=0}^{n} a_i = b_i \wedge [\ ] = [b_1, ; \dots; b_{k+1}]$$

Ce qui montre que $k = 0$ et $(a_0, \ldots, a_n) = (b_0, \ldots, b_n)$. Par obtenir la surjectivé, simplement note que par tout $m \in \mathbb{N}$, il y a $k \in \mathbb{N}$ tel que $\mathrm{tl}^k(m) = 0$ (parce que la suite $\{\mathrm{tl}^k(m)\}_{k\in\mathbb{N}}$ est strictement décroissante), alors

$$m = [\mathrm{hd}(m); \mathrm{hd}(\mathrm{tl}(m)); \ldots; \mathrm{hd}(\mathrm{tl}^k(m))] = [\mathrm{nth}(m, 0); \ldots; \mathrm{nth}(m, k)].$$

Exercice 14 (récurrence sur la suite des valeurs).

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

$$\begin{aligned} f(a_1, \ldots, a_p, 0) &= g(a_1, \ldots, a_p) \\ f(a_1, \ldots, a_p, x+1) &= h(\bar{a}, x, [f(\bar{a}, x); \ldots; f(\bar{a}, 0])). \end{aligned}$$

Il suffit de prouver que la fonctión $F(\bar{a}, x) = [f(\bar{a}, x); \ldots; f(\bar{a}, 0]$ est primitive recursive. On a

$$\begin{aligned} F(\bar{a}, 0) &= [f(\bar{a}, 0)] = g(\bar{a})) :: 0 \\ F(\bar{a}, x+1) &= f(\bar{a}, x+1) :: F(\bar{a}, x) = h(\bar{a}, x, F(\bar{a}, x)) :: F(\bar{a}, x) \end{aligned}$$

On a donc que $f(\bar{a}, x) = \mathrm{hd}(F(\bar{a}, x)$

(2) Montrer que la fonction $\mathrm{nthl}(l, i)$ qui a associe la suite codée par l à partie du $i + 1$-ième élément (0 sinon), de la fonction $\mathrm{nth}(l, i)$ qui à associe le $i+1$-ième élément de la suite codée par l, sont récursives primitives.

$$\begin{aligned} &\mathrm{nthl}(l, 0) = l && \mathrm{nth}(l, 0) = \mathrm{hd}(l) \\ &\mathrm{nthl}(l, i+1) = \mathrm{tl}(\mathrm{nthl}(l, i)) && \mathrm{nth}(l, i+1) = \mathrm{hd}(\mathrm{nthl}(l, i)) \end{aligned}$$

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

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

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

$$\begin{aligned} f(a_1, \ldots, a_p, 0) &= g(a_1, \ldots, a_p) \\ f(a_1, \ldots, a_p, x+1) &= h(\bar{a}, x, f(\bar{a}, p_1(x)), \ldots, f(\bar{a}, p_k(x)) \end{aligned}$$

est récursive primitive. On peut poser

$$\begin{aligned} f(\bar{a}, x+1) = h\Big(&\bar{a}, x, \mathrm{nth}([f(\bar{a}, 0); \ldots; f(\bar{a}, x)], x - p_1(x)), \\ &\ldots, \mathrm{nth}([f(\bar{a}, 0); \ldots; f(\bar{a}, x)], x - p_k(x))\Big) \end{aligned}$$

Exercice 15 (récurrence sur les listes).

(1) Montrer que f est récursive primitive

$$\begin{aligned} f(\bar{a}, []) &= g(\bar{a}) \\ f(\bar{a}, x :: l) &= h(\bar{a}, x, l, f(\bar{a}, l)). \end{aligned}$$

On peut poser

$$f(\bar{a}, y) = h(\bar{a}, \mathrm{hd}(y), \mathrm{tl}(y), f(\bar{a}, \mathrm{tl}(y)))$$

f est bien définie puisque la fonction liste est bijective.
mem

$$\mathrm{mem}(a, []) = 0$$
$$\mathrm{mem}(a, x :: l) = \chi_=(x, a)\,\mathrm{mem}(a, l)$$

@

$$@(l', []) = l'$$
$$@(l', x :: l) = x :: (l@l')$$

length

$$\mathrm{lg}([]) = 0$$
$$\mathrm{lg}(x :: l) = \mathrm{lg}(l) + 1$$

(2) Montrer que si f est rp, alors la fonction $\mathrm{map}(f)$ qui a $l = [\bar{u}]$ associe $[f(\bar{a}, u_1); \ldots; f(\bar{a}, u_p)]$

$$\mathrm{map}_f([]) = 0;$$
$$\mathrm{map}_f(x :: l) = f(\bar{a}, x) :: \mathrm{map}_f(l)$$

(3) concat

$$\mathrm{concat}([\ ]) = [\ ];$$
$$\mathrm{concat}(x :: l) = x :: [\mathrm{nth}(l, 0); \ldots, \mathrm{nth}(l, \mathrm{length}(l))]$$

subst

$$\mathrm{subst}([\ ], k, v) = [\ ];$$
$$\mathrm{subst}(x :: l, k, v) = \begin{cases} x :: \mathrm{subst}(l) & \text{si } x \neq v \\ \mathrm{concat}([k, \mathrm{subst}(l)] & \text{si } x = v \end{cases}$$

Exercice extra (Codage des listes par décomposition en nombres premiers) On note $\mathcal{S}$ l'ensembles des suites finies d'entiers. La fonction de codage des listes seq : $\mathcal{S} \to \mathbb{N}$ associe à chaque suite $(x_1, \ldots, x_k)$ la valeur suivante

$$\mathrm{seq}(x_1, \ldots, x_k) = p_0^k p_1^{x_1} \cdots p_k^{x_k}$$

envoyant la suive vide à 1.

(1) Montrer que ce codage est injectif mais pas surjectif. L'injectivité est claire par le théorème fondamental de l'arithmetique. Il n'y a pas de suite envoyée a 3, par exemple.
(2) Montrer que la fonction qui à (x, n) asoccie l'exposant de p_n dans la décomposition en facteurs premiers de x est récursive primitive.

$$\exp(x, n) = (\mu k \leq x)(p_n^{k+1} \nmid x)$$

(3) En déduire que
  (a) Il existe une fonction récursive primitive qui calcule le n-ième élément d'une suite representé par x, quant x représente une suite de longueur supérirure ou égale à n.
  Prends $\exp(x, n)$
  (b) Il existe une fonction rp qui calcule la longueur de la suite codée par x.
  Prends $l(x) = \exp(n, 0)$.
  (c) La fonction caractéristique de l'ensemble C des codes de suites est récursive primitive.
  On a $x \in A$ ssi $x \neq 0$ et $(x = 1 \vee 2|x)$.

(4) Montrer qu'il existe une fonction récursive primtive qui, à deux entiers $n = \mathrm{seq}(x_1, \dots, x_k)$ et $m = \mathrm{seq}(y_1, \dots, y_h)$ codant des suites renvoi le nombre représentant la concaténation des deux listes $\mathrm{seq}(\bar{x}, \bar{y})$.
Prends $\mathrm{concat}(n, m) = \mathrm{seq}(\exp(n, 1) \dots, \exp(n, k), \exp(m, 1), \dots, \exp(m, h))$

Exercice 16 (récurrence avec substitution de paramètre). C'est le schéma

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

(1) Montrer que la fonction F est RP

$$\begin{aligned} F(p, a, 0) &= g(\gamma^p(a)) \\ F(p, a, x+1) &= h(\gamma^{p-(x+1)}(a), x, F(p, a, x)). \end{aligned}$$

On voit que

$$F(p, a, x+1) = h(\mathrm{nth}([a; \gamma(a), \dots, \gamma^p(a), p - (x+1)]), x, F(p, a, x))$$

(2) Montrer que

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

et en déduire que f est récursive primitive.
Par récurrence sur x (on suppose que $x \leq p$ toujours)

$$\begin{aligned} F(p, a, 0) &= g(\gamma^p(a)) \\ F(p, a, x+1) &= h(\gamma^{p-(x+1)}(a), x, F(p, a, x)) \\ &= h(\gamma^{p-(x+1)}(a), x, f(\gamma^{p-x}(a), x)) \\ &= f(\gamma^{p-(x+1)}(a), x+1) \end{aligned}$$

On peut déduire que $f(a, x) = F(x, a, x)$.

(3) Application: montrer que la fonction $\mathrm{inc} : \mathbb{N}^2 \to \mathbb{N}$ qui à i et $l = [a_0; \dots; a_i; \dots; a_n]$ associe $[a_0; \dots; a_i + 1; \dots; a_n]$, est récursive primitive.
On peut poser

$$\begin{aligned} f(l, 0) &= (\mathrm{hd}(l) + 1) :: \mathrm{tl}(l) \\ f(l, i+1) &= \begin{cases} f(\mathrm{tl}(l), i) & \text{si } i \leq n \\ l & \text{sinon} \end{cases} \end{aligned}$$

Exercice 17 (récurrence double sans imbrication). Montrer que la fonction f définie par

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

est récursive primitive.
On peut utiliser le codage des couples ($t = \langle x, y \rangle$), et le schéma de recurrence sur la suite des

valeurs pour écrire f de la façon suivante

$$f(t) = \begin{cases} b & \text{si } \pi_2^1(t) = 0 \\ a & \text{sinon et } \pi_2^2(t) = 0 \\ h\Big(\pi_2^1(t) - 1, \pi_2^2(t) - 1, & \\ \text{nth}\Big([f(0); f(1); \ldots; f(\alpha(\pi_2^1(t), \pi_2^2(t) - 1)], \alpha(\pi_2^1(t) - 1, \pi_2^2(t) - 1)\Big), & \\ \text{nth}\Big([f(0); f(1); \ldots; f(\alpha(\pi_2^1(t), \pi_2^2(t) - 1)], \alpha(\pi_2^1(t), \pi_2^2(t) - 1)\Big)\Big), & \text{sinon} \end{cases}$$

Fonction d'Ackermann

$$\begin{aligned} \mathrm{Ack}(0, x) &= x + 2 \\ \mathrm{Ack}(1, 0) &= 0 \\ \mathrm{Ack}(n + 2, 0) &= 1 \\ \mathrm{Ack}(n + 1, x + 1) &= \mathrm{Ack}(n, \mathrm{Ack}(n + 1, x)) \end{aligned}$$

Exercice 18. Montrer que chaque fonction $\mathrm{Ack}_n(x) = \mathrm{Ack}(n, m)$ est récursive primitive et strictement croissante. Expliciter Ack_n, pour $n = 1, 2, 3$. On procéde par récurrence sur n. Si $n = 0, 1$, est evident que Ack_n est primitive récursive. On suppose que Ack_n est primtive récursive pour $n \geq 2$. On note que

$$\begin{aligned} \mathrm{Ack}_{n+1}(0) &= 1 \\ \mathrm{Ack}_{n+1}(x + 1) &= \mathrm{Ack}_n(\mathrm{Ack}_{n+1}(x)) \end{aligned}$$

Par hypothése de récurrence, $\mathrm{Ack}_{n+1} = \mathrm{Rec}(1, \mathrm{Ack}_n \circ \pi_2^2) \Rightarrow \mathrm{Ack}_{n+1}$ est récursive primitive. On a aussi

$$\begin{aligned} \mathrm{Ack}_1(x) &= 2x \\ \mathrm{Ack}_2(x) &= 2^x \\ \mathrm{Ack}_3(x) &= \underbrace{2\,\hat{}\cdots\hat{}\,2}_{x-\text{fois}}\hat{}\,x \end{aligned}$$

Le fait que la fonction soit strictement croissante découle d'une application immédiate de la récurrence sur n.

Exercice 19 (fonction d'Ackermann)

(1) Vérifier qu'il existe bien une et une seule fonction de $\mathbb{N}^2 \to \mathbb{N}$ vérifiant les équations de la fonction d'Ackermann. On considére la recurrence

$$\mathrm{Ack}_{n+1}(x + 1) = \mathrm{Ack}_n(\mathrm{Ack}_{n+1}(x)),$$

et on note que, si $>_{lex}$ denole l'ordre lexicographique sur $\mathbb{N}^2$, on a

$$\begin{aligned} (n + 1, x + 1) &>_{lex} (n + 1, x) \\ (n + 1, x + 1) &>_{lex} (n, \mathrm{Ack}_{n+1}(x)) \end{aligned}$$

On peut en déduire que, pour calculer la valeur $\mathrm{Ack}_{n+1}(x + 1)$, on a besoin des valeurs que Ack prends en couples strictement mineurs que $(x+1, n+1)$ (selon $>_{lex}$). Puis que $(\mathbb{N}^2, >_{lex})$ est un ensemble bien ordonné, il ensuit que l'ensemble de couples nécessaires pour calculer $\mathrm{Ack}_{n+1}(x + 1)$ est fini. Pourtant, Ack est bien défini sur $\mathbb{N}^2$, et elle est «intuitivement calculable».

(2) Montrer que

$$\forall n \in \mathbb{N}\ \forall x > 0\, \mathrm{Ack}_{n+1}(x) = \mathrm{Ack}_n^x(\mathrm{Ack}_{n+1}(0)))$$

et vérifier les expressions des fonctions Ack_1, Ack_2, Ack_3. Par récurrence sur x. Si $x = 0$, c'est trivial. Par le cas $x + 1$, on a

$$\begin{aligned} \mathrm{Ack}_{n+1}(x+1) &= \mathrm{Ack}_n(\mathrm{Ack}_{n+1}(x)) \text{ par définition} \\ &= \mathrm{Ack}_n(\mathrm{Ack}_n^x(\mathrm{Ack}_{n+1}(0))) \text{ HI} \\ &= \mathrm{Ack}_n^{x+1}(\mathrm{Ack}_{n+1}(0))) \end{aligned}$$

(3) Vérifiez que chacune des fonctions Ack_n a une définition en utilisant exactement n instances du schéma de définition par itération. Les formes explicites de $\mathrm{Ack}_1, \mathrm{Ack}_2, \mathrm{Ack}_3$, sont dans l'exercice 18. Cela découle directement de la définition, et par récurrence sur n, en notant que

$$\begin{aligned} \mathrm{Ack}_{n+1}(0) &= 1 \\ \mathrm{Ack}_{n+1}(x+1) &= \mathrm{Ack}_n(\mathrm{Ack}_{n+1}(x)) \end{aligned}$$

Alors, si $\mathrm{Ack}_n \in \mathcal{C}_n$, $\mathrm{Ack}_{n+1} \in \mathcal{C}_{n+1}$.

(4) Montrer que $\mathrm{Ack}_n(x) > x$.
Par récurrence sur n. Si $x > 0$,

$$\begin{aligned} \mathrm{Ack}_0(x) &= x + 2 > x \\ \mathrm{Ack}_1(x) &= 2x > x \end{aligned}$$

Si $n \geq 2$ et on suppose que par tout $x > 0$, $\mathrm{Ack}_n(x) > x$,

$$\mathrm{Ack}_{n+1}(1) = \mathrm{Ack}_n(\mathrm{Ack}_{n+1}(0)) = \mathrm{Ack}_n(1) > 1$$

Si $x > 1$, puisque Ack est strictement croissante, $\mathrm{Ack}_{n+1}(x) > 1 \neq 0$, et on peut apliquer récurrence sur x,

$$\begin{aligned} \mathrm{Ack}_{n+1}(x+1) &= \mathrm{Ack}_n(\mathrm{Ack}_{n+1}(x)) \\ &\geq \mathrm{Ack}_{n+1}(x) + 1 \text{ (HI1)} \\ &> x + 1 \text{ (HI2)} \end{aligned}$$

(5) En déduire que pour tout entier m, Ack_m est strictement croissante.
Cela a déjà été démontré lors de l'exercice précédent.

(6) Déduire de la question 4, que, à partir de 2, Ack est croissante au sens large sur son premier argument, le second étant fixé:

$$\forall x \geq 2\ \forall n \in \mathbb{N}\ \ \mathrm{Ack}(n, x) \leq \mathrm{Ack}(n+1, x).$$

On a

$$\mathrm{Ack}_{n+1}(x) = \mathrm{Ack}_n(\underbrace{\mathrm{Ack}_{n+1}(x-1)}_{\geq x}) \geq \mathrm{Ack}_n(x)$$

(7) Montrer que $\forall k, n \in \mathbb{N}\, \mathrm{Ack}_n^k \in \mathcal{C}_n$.
C'est clair en vu de l'exercice 19.3 et puisque $\mathcal{C}_n$ est clos par composition.

(8) Montrer que $\forall k, n \in \mathbb{N}\,\mathrm{Ack}_n^k(x) \leq \mathrm{Ack}_{n+1}(x+k)$. Par récurrence sur k, le cas $k=0$ est trivial, puis

$$\begin{aligned}\mathrm{Ack}_n^{k+1}(x) &= \mathrm{Ack}_n(\mathrm{Ack}_n^k(x)) \\ &\leq \mathrm{Ack}_n(\mathrm{Ack}_{n+1}(x+k)) \text{ HI} \\ &= \mathrm{Ack}_{n+1}(x+k+1) \text{ def}\end{aligned}$$

(9) Montrer para récurrence sur la définition de l'ensemble des fonctions récursives primitives que si $f \in \mathcal{C}_n$, alors $\exists k\,\mathrm{Ack}_n^k$ domine f.
Il est facile de voir que les fonctions base sont dominées par $\mathrm{Ack}_3(x)$.
Si $h, g_1, \ldots, g_m \in \mathcal{C}_n$,$h(\bar{x}) \leq \mathrm{Ack}_n^k \sup(\bar{x}, K)$ et $g_i(\bar{x}) \leq \mathrm{Ack}_n^{k_i} \sup(\bar{x}, K_i)$, on posse $M = \sup(K_1, \ldots, K_m, K)$, $l = \sup_i k_i$, et $M(\bar{x}) = \sup(\bar{x}, M)$.

$$\begin{aligned}h(g_1(\bar{x}), \ldots, g_m(\bar{x})) &\leq \mathrm{Ack}_n^k \sup_i(g_i(\bar{x}), K) \\ &\leq \mathrm{Ack}_n^k \sup_i(\mathrm{Ack}_n^{k_i} \sup(\bar{x}, K_i), K) \\ &\leq \mathrm{Ack}_n^k \sup_i(\mathrm{Ack}_n^{k_i}(M(\bar{x}))) \\ &= \mathrm{Ack}_n^k(\mathrm{Ack}_n^l(M(\bar{x}))) \\ &= \mathrm{Ack}_n^{k+l} \sup(\bar{x}, M)\end{aligned}$$

Maintenant, si $g(\bar{x}) \leq \mathrm{Ack}_n^{k_1} \sup(\bar{x}, N_1)$, et $h(\bar{x}, y, z) \leq \mathrm{Ack}_n^{k_2} \sup(\bar{x}, y, z, N_2)$, la fonction obtenu par récurrence primitive $f \in \mathcal{C}_{n+1}$ satisfait

$$f(\bar{x}, y) \leq \mathrm{Ack}_n^{k_1+k_2y}(\sup(\bar{x}, y, N_1, N_2)$$

On démontre par récurrence sur y,

$$\begin{aligned}f(\bar{x}, 0) &= g(\bar{x}) \leq \mathrm{Ack}_n^{k_1} \sup(\bar{x}, N_1) \\ f(\bar{x}, y+1) &= h(\bar{x}, y, f(\bar{x}, y)) \\ &\leq \mathrm{Ack}_n^{k_2}(\sup(\bar{x}, y, f(\bar{x}, y), N_2) \\ &\leq \mathrm{Ack}_n^{k_2}(\sup(\bar{x}, y, \mathrm{Ack}_n^{k_1+k_2y} \sup(\bar{x}, y, N_1, N_2), N_2) \\ &= \mathrm{Ack}_n^{k_2}(\mathrm{Ack}_n^{k_1+k_2y} \sup(\bar{x}, y, N_1, N_2)) \\ &= \mathrm{Ack}_n^{k_1+k_2(y+1)} \sup(\bar{x}, y, N_1, N_2) \\ &\leq \mathrm{Ack}_{n+1}(\sup(\bar{x}, y, N_1, N_2) + k_1 + k_2y)\end{aligned}$$

Cette dernière fonction est une composition entre fonctions C_{n+1} et pourtant est dominé par certain Ack_{n+1}^l.

(10) Montrer que Ack_n^k est dominée par Ack_{n+1}.
Note que si $y > 0$, $\mathrm{Ack}_{n+1}(y) \geq \mathrm{Ack}_1(y) = 2y$, et on peut déduire que si $x > 2k$, $\mathrm{Ack}_{n+1}(x-k) \geq 2x - 2k > x$. Alors, par tout $x > 2k$,

$$\begin{aligned}&\mathrm{Ack}_{n+1}(x-k) > x \\ \Rightarrow &\mathrm{Ack}_n^k(\mathrm{Ack}_{n+1}(x-k)) > \mathrm{Ack}_n^k(x) \\ \Rightarrow &\qquad \mathrm{Ack}_{n+1}(x) > \mathrm{Ack}_n^k(x) \text{ (ex 19.2)}\end{aligned}$$

Ce qui montre que Ack_n^k est dominée par Ack_{n+1}.

(11) En déduire que si $f \in \mathcal{C}_n$, alors Ack_{n+1} domine f.
Si $f \in \mathcal{C}_n$, $\exists k$ tel que f est dominée par Ack_n^k, et par l'exercice précedent, Ack_n^k est dominée par Ack_{n+1}. De plus, $\mathrm{Akc}_{n+1} \notin \mathcal{C}_n$.

(12) En déduire que la fonction d'Ackermann n'est pas récursive primitive. Montrer que la fonction diagonale $\mathrm{Ack}(n,n)$ domine toutes les fonctions récursives primitives.
Si $\mathrm{Ack}(n,n) \in \mathcal{C}_k$,

$$\exists N\ \forall n > N\ \ \mathrm{Ack}(n,n) \leq \mathrm{Ack}_k(n),$$

ce qui est imposible si $n > N, k$. Si f est primitive récursive, $f \in \mathcal{C}_n$ par certain n, donc en utilisant les exercise précedents, sauf finis valeurs de $\bar{x}$

$$f(\bar{x}) \leq \mathrm{Ack}_n^k(\sup(\bar{x})) \leq \mathrm{Ack}_{n+1}(\sup(\bar{x})) \leq \mathrm{Ack}_{\sup(\bar{x})}(\sup(\bar{x}))$$