Fonctions Récursives

Approche basée sur les fonctions définies sur les nombres naturels pour caractériser celles qui sont calculables par une procédure effective.

Fonctions calculables connues :
- les fonctions de base dont un algorithme est connu pour les calculer sur une représentation ;
- la composition finie de fonctions de base ;
- la récursion (…en partie).

Exemple : Exemples de fonctions récursives :
- + et x sur les réels,
- la factorielle,
- la racine carré d'un réel positif par la méthode de Newton

Fonctions récursives primitives

def Les fonctions récursives primitives simples sont :
- la fonction $\overline{0}$,
- les fonctions de projection
$\pi_i^k = (\lambda n_1 \ldots n_k \;\; n_i)$,
- la fonction successeur $(\lambda n \;\; (\sigma \;\; n))$.

def La composition de la fonction g à l arguments et des fonctions $h_1, \ldots, h_l$ à k arguments est la fonction f à k arguments définie par :
$(\lambda n_1 \ldots n_k \; (g \; (h_1 \; n_1 \ldots n_k) \ldots (h_l \; n_1 \ldots n_k)))$.

def La récursion primitive à partir des fonctions g à k arguments et h à $k+2$ arguments est la fonction f à $k+1$ arguments définie par :

$$\begin{array}{ll} (\lambda & n_1 \ldots n_k m \; (if \; (m = (\sigma \; m')) \\ & then \; (h \; (f \; n_1 \ldots n_k \; m') \; n_1 \ldots n_k \; m') \\ & else \; (g \; n_1 \ldots n_k))) \end{array}$$

Fonctions récursives primitives (suite)

def Les fonctions récursives primitives sont définies inductivement par :
- les fonctions récursives primitives simples,
- la composition et la récursion primitive.

Exemple : Les fonctions constantes $\overline{j}$ sont des fonctions primitives récursives :
$\overline{j} = \underbrace{(\sigma\ (\sigma \ldots (\sigma\ \overline{0})))}_{\text{j fois}}.$

Exemple : L'addition, la soustraction et la multiplication sur les naturels sont des fonctions récursives primitives.

Exemple : La factorielle est une fonction récursive primitive.

Exemple : Le produit borné par m d'une fonction $(\lambda n_1 \ldots n_k i\ (g\ n_1 \ldots n_k\ i))$ défini par le produit de g pour les m premières valeurs de i est une fonction récursive primitive.

Les prédicats récursifs primitifs

def Un prédicat à k arguments est un sous-ensemble de $\mathbb{N}^k$.

def La fonction caractéristique d'un prédicat $P \subseteq \mathbb{N}^k$ est la fonction f_P de $\mathbb{N}^k$ dans $\{0, 1\}$ définie par :

$$(f_P\, n_1 \ldots n_k) = \begin{array}{l} \overline{0} \text{ si } (n_1, \ldots, n_k) \notin P \\ \overline{1} \text{ sinon} \end{array}$$

def Un prédicat est récursif primitif si sa fonction caractéristique est récursive primitive.

Exemple : Le prédicat *nonzero* qui est vrai sauf si son argument est l'entier 0 a pour fonction caractéristique $f_{\overline{\emptyset}}$:

$$\begin{array}{ll} (\lambda m & (if\ (m = (\sigma\ m')) \\ & then\ (\pi_3^3\ (f_{\overline{\emptyset}}\ m')\ m'\ \overline{1}) \\ & else\ \overline{0})) \end{array}$$

Exemple : Écrire les fonctions caractéristiques (récursives primitives) des prédicats *et*, *ou* et *non*.

Les prédicats récursifs primitifs (suite)

def La quantification bornée universelle teste si un prédicat est vrai pour toutes les valeurs d'un de ses arguments inférieures à une borne.

prop.: La quantification bornée universelle est un prédicat récursif primitif.

Soit f_P la fonction caractéristique du prédicat P à $k+1$ arguments. Alors la fonction caractéristique (récursive primitive) de la quantification bornée de P pour le $k+1$ argument $\forall^{\leq}_{P}$ est définie par :

$$
\begin{array}{l}
(\lambda n_1 \ldots n_k m\ (if\ (m = (\sigma\ m'))\\
then\ (fois\ \ (\pi_1^{k+2}\ (\forall^{\leq}_{P}\ n_1 \ldots n_k\ m')\ n_1 \ldots n_k\ m')\\
\qquad\qquad\quad (f_P\ n_1 \ldots n_k\ m'))\\
else\ (\pi_1^{k+2}\ \overline{1}\ n_1 \ldots n_k\ m')))
\end{array}
$$

Les prédicats récursifs primitifs (suite)

def La quantification bornée existentielle teste si un prédicat est vrai pour des valeurs d'un de ses arguments inférieures à une borne.

prop.: La quantification bornée existentielle est un prédicat récursif primitif.

def Pour une borne, la valeur de la fonction de minimisation bornée d'un prédicat (pour un de ses arguments) est le plus petit nombre naturel inférieur à la borne tel que le prédicat soit vrai :

$$\mu_P i \leq m = \begin{cases} \text{le plus petit } i \leq m \text{ tel que} \\ (n_1, \ldots, n_k, i) \in P \\ 0 \text{ si il n'existe pas de tel } i \end{cases}$$

prop.: La minimisation bornée est une fonction récursive primitive.

Les fonctions μrécursives

Th.: Il existe des fonctions calculables qui ne sont pas primitives récursives.

def La valeur de la fonction de minimisation non bornée d'un prédicat (pour un de ses arguments) est le plus petit nombre naturel tel que le prédicat soit vrai.

remarque : Dans le cas d'un prédicat toujours faux, la minimisation non bornée n'est pas calculable car ne termine jamais.

def Un prédicat P est sûr si pour tout $n_1, \ldots, n_k$, il existe un i tel que $(n_1, \ldots, n_k, i) \in P$.

def Les fonctions μrécursives sont celles obtenues par induction sur les fonctions récursives primitives simples, la composition, la récursion primitive et la minimisation non bornée des prédicats sûrs.

Fonctions μrécursives et Machines de Turing

2 approches pour la définition des fonctions calculables par des procédures effectives : les fonctions μrécursives et les fonctions calculables par machine de Turing.

Question : Les fonctions μrécursives sont-elles exactement les fonctions calculables par une machine de Turing ?

Fonctions μrécursives sur les naturels	Machine de Turing sur les caractères

Problème de la représentation effective des nombres par des caractères et inversement : nécessité de procédures effectives de codage.

Nombres et caractères

lem.: Il existe une représentation effective des nombres naturels par des chaînes de caractères.

pr. inf.: Le codage binaire des naturels.

lem.: Il existe une représentation effective des chaînes de caractères par des nombres naturels.

def (la Gödelisation) La représentation d'une chaîne de caractère par un nombre de Gödel.

Exemple : Soit $\mathcal{A}$ un alphabet. Nous définissons une bijection gd de $\mathcal{A}$ dans $\{1, \ldots, |\mathcal{A}|\}$. Pour tout mot $w = \sigma_0 \ldots \sigma_l \in \mathcal{A}^*$, le nombre de Gödel de w est défini par :

$$Gd(w) = \Sigma_{i=0}^{l}(1 + |\mathcal{A}|)^{l-i} gd(w_i).$$

Fonctions μrécursives $\equiv$ machines de Turing

Th.: Toute fonction μrécursive est calculable par une machine de Turing.

pr. inf.: En adoptant un codage des entiers narurels, les fonctions primitives récursives simples, leurs compositions, leurs récursions primitives et leurs minimisations non bornées sûres sont calculables par une machine de Turing.

Th.: Toute fonction calculable par une machine de Turing est μrécursive.

pr. inf.: Soient une machine de Turing M sur un alphabet $\mathcal{A}$ calculant une fonction f_M de $\mathcal{A}^*$ dans $\mathcal{A}^*$ et une gödédisation Gd pour $\mathcal{A}^*$. Alors il existe une fonction μrécursive f telle que $(\forall w \in \mathcal{A}^*(f_M(w) = Gd^{-1}(f(Gd(w)))))$.

Fonctions partielles

def Une fonction partielle f de $\mathcal{A}^*$ dans $\mathcal{A}^*$ est calculée par une machine de Turing M si,
- pour tout mot d'entrée w tel que f soit définie, M s'arrête toujours dans une configuration ou se trouve $f(w)$ sur le ruban ;
- pour tout mot d'entrée w tel que f ne soit pas définie, M ne s'arrête pas ou s'arrête sur une configuration conventionnelle.

def Une fonction partielle f de $\mathbb{N}$ dans $\mathbb{N}$ est μrécursive si elle est définie par induction sur les fonctions récursives primitives simples, la composition, la récursion primitive et la minimisation non bornée.

Th.: Une fonction partielle est μrécursive si et seulement si elle est calculable par une machine de Turing.