Fonctions récursives

Trois grands modèles :

- Machines de Turing (Turing 36, Post 36)
- Fonctions récursives (Gödel 31, Kleene ~40, Ackermann ~40)
- λ-calcul (Church ~30)

  $\rightsquigarrow$ λ-calculs typés (Church ~40...)

Fonctions récursives

Définition axiomatique et pas comme langage
$\rightsquigarrow$ règles de génération des fonctions

Dans la suite : $\mathcal{F}_k = \{f \mid f : \mathbb{N}^k \to \mathbb{N}\}$

Fonctions récursives primitives

Constantes

$$C_{k,c} \in \mathcal{F}_k, \quad C_{k,c}(x_1, \ldots, x_k) = c$$

Successeur

$$S \in \mathcal{F}_1, S(x) = x + 1$$

Projections

$$\pi_{k,i} \in \mathcal{F}_k, \quad \pi_{k,i}(x_1, \ldots, x_i, \ldots, x_k) = x_i$$

Fonctions récursives primitives

Schéma de Composition :

- f à n arguments
- $g_1, \ldots, g_n$ à m arguments

$\rightsquigarrow$ h à m arguments

$$h(x_1, \ldots, x_m) = f(g_1(x_1, \ldots, x_m), \ldots, g_n(x_1, \ldots, x_m))$$

$h = \textit{Comp}_{n,m}(f, g_1, \ldots, g_n)$

Schéma de récursion primitive

- b à n arguments
- h à $n+2$ arguments

$\leadsto$ f à $n+1$ arguments

$$\begin{array}{rcl} f(0, x_1, \ldots, x_n) & = & b(x_1, \ldots, x_n) \\ f(k+1, x_1, \ldots, x_n) & = & h(k, x_1, \ldots, x_n, \underbrace{f(k, x_1, \ldots, x_n)}) \end{array}$$

$f = \textit{Rec}(b, h)$

Ensemble $\mathscr{RP}$ des fonctions récursives primitives :

- Constantes
- Projections
- Successeur
- Clos par composition
- Clos par récursion primitive

Ex. : addition ? multiplication ?

Fonctions récursives primitives

« Prédicat » récursif primitif : fonction à valeur dans $\{0, 1\}$

Ex. : $<, >, \leq, \geq, =, \neq, \ldots$

Ex. : composition avec $\neg, \wedge, \vee$

Bijection réc. prim $\alpha : \mathbb{N}^2 \rightarrow \mathbb{N}$ avec réciproques $\alpha_1^{-1}, \alpha_2^{-1}$ réc. prim.

$$\begin{array}{rcl} \alpha(\alpha_1^{-1}(x), \alpha_2^{-1}(x)) & = & x \\ \alpha_1^{-1}(\alpha(x,y)) & = & x \\ \alpha_2^{-1}(\alpha(x,y)) & = & y \end{array}$$

$\rightsquigarrow$ Codage des n-uplets $\qquad (x, y)$ codé $\alpha(x, y)$

Fonctions récursives primitives

Hiérarchie de Grzegorczyk

Niveaux de fonctions

Par niveau : croissance plus rapide que niveaux inférieurs

$$\begin{array}{rcl} \xi_0(x) & = & x + 1 \\ \xi_{n+1}(0) & = & \xi_n(1) \\ \xi_{n+1}(x+1) & = & \xi_n(\xi_{n+1}(x)) \end{array}$$

$$\begin{array}{llcl} n = 1 & \xi_1(x) & = & x + 2 \\ n = 2 & \xi_2(x) & = & 2x + 3 \\ n = 3 & \xi_3(x) & = & 2^{x+3} - 3 \\ n = 4 & \xi_4(x) & = & \ldots \end{array}$$

Fonctions récursives primitives

Ackermann-Péter :

$$\begin{array}{lcl} \mathit{Ack}(0, x) & = & x + 1 \\ \mathit{Ack}(x+1, 0) & = & \mathit{Ack}(x, 1) \\ \mathit{Ack}(x+1, y+1) & = & \mathit{Ack}(x, \mathit{Ack}(x+1, y)) \end{array}$$

Totale ? $\qquad$ équiv. : « termine ? »

Qq. ex. . .

Notation : $A_n(x) = \mathit{Ack}(n, x)$ $\qquad \in \mathscr{RP}$

Et *Ack* ?

Fonctions récursives

Schéma de minimisation :

- g à $n + 1$ arguments
- $\rightsquigarrow$ f à n arguments

$$f(x_1, \ldots, x_n) = \min\{k \mid g(k, x_1, \ldots, x_n) = 0\}$$

$f = \mathit{Min}(g)$

Si pas de $\min$ $\rightsquigarrow$ indéfini

Fonctions récursives

Ensemble $\mathscr{R}$ des fonctions récursives :

- Constantes
- Projections
- Successeur
- Clos par composition
- Clos par récursion primitive
- Clos par minimisation

Ex. : pred non déf. en 0 ? nulle part ?

Fonctions récursives

Fonctions récursives totales = fonctions récursives définies partout. . .

$$\mathscr{RP} \subset \mathscr{R}_{\text{tot}} \subset \mathscr{R}$$

Ensemble E récursif = χ_E récursive totale

Ensemble E récursivement énumérable = χ_E récursive définie sur tout E

Comparaison avec Turing-calculables ?

Fonctions récursives

Fonctions récursives totales = fonctions récursives définies partout. . .

$$\mathscr{RP} \subset \mathscr{R}_{\text{tot}} \subset \mathscr{R}$$

Ensemble E récursif = χ_E récursive totale

Ensemble E récursivement énumérable = χ_E récursive définie sur tout E

Récursive totale $\Rightarrow$ Turing-calculable